PERFORMANCE EVALUATION OF ASYNCHRONOUS MULTI-CHANNEL MAC PROTOCOLS FOR 802.11 WIRELESS NETWORKS

BY

ABDULLAH DEVENDIRAN

A Thesis Presented to the
DEANSHIP OF GRADUATE STUDIES

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS

DHAHRAN, SAUDI ARABIA

In Partial Fulfillment of the
Requirements for the Degree of

MASTER OF SCIENCE

In

COMPUTER NETWORKS

Mar 9, 2016

KING FAHD UNIVERSITY OF PETROLEUM & MINERALS
DHAHRAN 31261, SAUDI ARABIA

DEANSHIP OF GRADUATE STUDIES

This thesis, written by **ABDULLAH DEVENDIRAN** under the direction of his thesis adviser and approved by his thesis committee, has been presented to and accepted by the Dean of Graduate Studies, in partial fulfillment of the requirements for the degree of **MASTER OF SCIENCE IN COMPUTER NETWORKS**.

Thesis Committee

Dr. Tarek Sheltami (Adviser)

Dr. Yahya Esmail Osais (Member)

Marwan

Dr. Marwan Abu-Amara (Member)

for AM

Dr. Ahmed AlMulhem
Department Chairman

Dr. Salam A. Zummo
Dean of Graduate Studies

24/3/16

Date

Dedicated to King Fahd University of Petroleum and Minerals

ACKNOWLEDGMENTS

First and Foremost, all praises are due to Allah alone Who made this possible. I thank Him for His countless blessings and favours upon me. With His grace, all the good deeds are completed. And Peace and Blessings of Allah upon His chosen slave and Messenger, Muhammad, and those who follow him until the Last Day. And then, I am quite privileged to have Dr.Tarek Sheltami as my advisor. I thank him very much for his consistent support and guidance throughout this thesis. I thank my committee members Dr.Yahya Osais and Dr.Marwan Abu-Amara for their support and helpful inputs. I am highly indebted to Dr.Ashraf Mahmoud, for his basic DCF simulator code in Matlab and discussions related to it. I gratefully acknowledge that this work is a fundamental building block for my python simulator. Thanks to my wife, kids, and my family members who have been a constant source of support. I thank all of my friends -a quite a lot- who helped me in this challenging journey. Finally, Thanks to the Department of Computer Engineering, King Fahd University of Petroleum and Minerals for this great opportunity and also to the Kingdom of Saudi Arabia for the numerous endeavours they provide to students.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

THESIS ABSTRACT

NAME:	ABDULLAH DEVENDIRAN
TITLE OF STUDY:	PERFORMANCE EVALUATION OF ASYNCHRONOUS MULTI-CHANNEL MAC PROTOCOLS FOR 802.11 WIRELESS NETWORKS
MAJOR FIELD:	COMPUTER NETWORKS
DATE OF DEGREE:	Mar 9, 2016

Advances in physical layer techniques such as multi-rate transmission, smart antennas, etc., promise high data rate transmissions in wireless networks. However, the underlying MAC layer needs to improve in order to fully exploit these features, and to support high performance applications. By enabling concurrent transmissions over non-interfering channels, multi channel MAC protocols seek to maximize the network performance. Although the transceivers can switch between channels with ease, the available 802.11 protocol is designed only for a single channel. Some multi-channel MAC protocols are hard and impractical to implement in wireless networks, especially those based on tight global synchronization, and that lack broadcast support. In this work, we propose an asynchronous bidi-

rectional multi-channel MAC (ABMMAC), which is 802.11 compatible, low cost, and utilizes spectrum effectively, by using just a single half duplex transceiver. The proposed approach provides support for broadcast and is a logical extension of 802.11. A comparative evaluation against multi-channel protocols employing asynchronous mode of operation such as DCA, AMMAC, and BiMMAC is carried out. Simulation results show that the proposed MAC gives a better performance over its multi-channel variants and legacy 802.11 under small to big network sizes.

ملخص الرسالة

الاسم الكامل: عبد الله ديفيندران

عنوان الرسالة: تقييم أداء بروتوكولات MAC غير المتزامنة متعددة القنوات للشبكات اللاسلكية 802.11

التخصص: هندسة حاسب آلي

تاريخ الدرجة العلمية: 29 جمادي الأولى 1437هـ الموافق 9 مارس 2016

التقدم الكبير في تقنيات الطبقة الفيزيائية للشبكات اللاسلكية مثل الارسال بمعدلات متعددة و الهوائيات الذكية ... يبشر بسرعات عالية في الشبكات الـلاسلكية. لكن بالرغم من ذلك لابد من تعديل طبقة ال MAC للاستفادة من تلك التقنيات في زيادة كفاءة الشبكات اللاسلكية و القدرة على دعم التطبيقات ذات الأداء العالي . بروتوكولات MAC متعددة القنوات تعمل على زيادة كفاءة الشبكة بتمكين الإرسال المتزامن في القنوات غير المتداخلة. بالرغم من توفر المبادلات التى تتيح التنقل بين القنوات بسهولة إلا أنه تم تصميم بروتوكول 802.11 للعمل على قناه واحدة فقط. الطرق المتاحة لتعدد القنوات التي تتطلب التزامن الصارم بين طرفي الاتصال و كذلك التي لا تدعم الإرسال الجماعي غير مناسبة للشبكات اللاسلكية. هذه الدراسة تقدم MAC بروتوكول غير متزامن ثنائي الإتجاة و متعدد القنوات متوافق مع 802.11 قليل التكلفة و مناسب للمبدلات التي ترسل باتجاه واحد و يستخدم نطاق التردد بكفاءة. وكذلك يدعم الإرسال الجماعي. تم إجراء تقييم شامل و مقارنة مع البروتوكولات متعددة النطاقات التى تدعم الإرسال غير المتزامن مثل DCA و AMMAC و BiMMAC. النتائج اظهرت الكفاءة العالية للبروتوكول مقارنة بالبروتوكولات المشابهة و البروتوكول التقليدي 802.11 في الشبكات الصغيرة والكبيرة

CHAPTER 1

INTRODUCTION

Medium Access Control (MAC) layer significantly impacts the network throughput, as it co-ordinates the channel access between wireless nodes. Advances in physical layer techniques such as multi-rate transmission, multiple coding and modulation, smart antennas making use of spatial multiplexing, transmit diversity, interference nulling, etc., promise high data rate transmissions in wireless networks [2]. However, the MAC layer needs to improve in order to exploit these features, and to support high performance applications. By enabling concurrent transmissions over non-interfering channels, multi channel MAC protocols seek to maximize the network performance.

The IEEE standard [3] defines multiple channels for communication at the physical layer. 802.11 b, g specifies 3 orthogonal or non-overlapping channels which are 22MHz wide. Every fifth channel can be used effectively without overlap, for example, channels 1, 6, and 11. 802.11a specify 12 non-overlapping channels: 4 channels of 20MHz each in the U-NII upper, lower, and middle bands.

When orthogonal channels are used, concurrent transmissions can co-exist on multiple channels without interference to each other. Although a transceiver is able to switch easily between various channels, the available 802.11 protocol is designed for a single channel.

Research efforts in multi-channel follow various paradigms: dedicated control channel, split phase, common hopping, and parallel-rendezvous. Each category has its specific advantages and weaknesses. Global network-wide synchronization is practically impossible for 802.11 networks unless with application of external devices like GPS. Also, support for broadcast is paramount to support routing and ARP messages. Our contribution includes:

- Proposal of Asynchronous Bidirectional Multi-channel MAC (ABMMAC) using a single half-duplex transceiver, backward compatible with 802.11, and with support for broadcast.

- Comparative evaluation with other multi-channel protocols namely: Dynamic Channel Assignment (DCA), Asynchronous Multi-Channel MAC (AMMAC), Bidirectional Multi-Channel MAC (BiMMAC), and the legacy single channel 802.11 with its RTS-CTS variant.

The proposed asynchronous bidirectional multi-channel MAC is low cost, with efficient spectrum utilization using a single half duplex transceiver, and an asynchronous mode of operation. Simulation results show that the proposed MAC gives a higher performance over 802.11 and its multi-channel variants in small to big network sizes.

CHAPTER 2

BACKGROUND AND LITERATURE REVIEW

In designing a multi-channel MAC (medium access control), two challenges need to be addressed [4], [5]:

a) *Medium access mechanism:* How nodes negotiate to obtain a channel? For example, contending on a control channel [6] or Ad-hoc traffic Indication Message (ATIM) window [7].

b) *Channel selection algorithm:* How to choose from a pool of available data channels? Which algorithm will yield effective performance? For example, lowest numbered channel, random selection, soft reservation [8]. How long the channel will be used between the same pair of nodes? Such issues need to be addressed. Channel assignment can happen on a per packet basis as in [9] [10], link basis as in [7], flow basis as in [11], or component basis as in [12]. Also, channel assignment methods can be static, dynamic, or semi-dynamic

[13] [14].

Using multiple transceivers per node has been investigated in several works. In [15], the authors use two transceivers with one intended for transmission operating in fast mode, and other intended for reception working in slow mode. In [16], several continuous data frames are sent on the earlier agreed channel skipping channel negotiations, lessening overhead on the control channel and allow more transmissions. In [17], the authors use a single base channel by default, and switch to other channels as the load increases. In [18], the authors use several transceivers to increase performance. Using more than one transceiver increases cost, size, and energy consumption of a node.

Though multi-channel protocols perform significantly better than a single channel type, they introduce new kind of problems:

- **Multi-channel hidden terminal problem:** While two nodes involve in data transfer on a certain channel, they will miss the control packets exchange sent on other channels. Due to missing information on channels status, the two nodes may inadvertently choose a busy channel and start a data exchange, causing a collision.

- **Missing receiver problem:** Control packets sent on a certain channel to an intended node fail, as the node is busy in another channel either sending or receiving.

- **Global knowledge of topology, traffic requirements:** Synchronization based techniques need global information on topology and traffic such as global time slot synchronization as in TDMA, or pre-distribution of code as in CDMA. Synchronization can be achieved by exchange of timestamps and broadcast. Since broadcast messages are sent without any reservation of the channel, it is highly susceptible to collisions and heavy load situations make collisions even more. In synchronizing between different node clocks, due to driftness and other issues, achieving less than 1ms accuracy is extremely difficult. Unless using external devices like GPS, it is better to avoid systems using synchronization.

- **Other:** Other issues in multi-channel include: how to decide if the current receiver is on the control channel, when to start negotiations, how to compensate the missed control exchanges when the nodes were involved in data transfer, how to reduce channel status overhead, how to balance the load among the channels, etc.,[19]. Problems such as Information Asymmetry, Flow in the Middle [20] are also present.

Proposed solutions to encounter multi-channel hidden terminal include a dedicated control channel as in DCA [6], or scanning all the channels after the data transfer and update channel status table [21]. Wormsbecker et al., [8] note that, a careful channel selection strategy can avoid multi-channel hidden problem. Giving preference to a past-used-channel is a safe option than randomly selecting a channel or selecting a least numbered channel. In [22], neighbors can interrupt the control transmissions. If a data channel in use is selected inadvertently , the neighbors issue invalid signal to stop the on-going negotiation.

2.1 Classification of multi-channel MAC protocols

Existing wireless MAC protocols are classified on many aspects like single transceiver or multi-transceiver, synchronous or asynchronous, fixed assignment or dynamic assignment, contention based or schedule based, single channel or multi-channel, etc., Parameters like signaling technique, type of architecture, sharing mode, access mode are also considered. Research efforts in multi-channel follow various paradigms: dedicated control channel [16],[23],[24],[25],[22], split phase [7],[26],[27],[28],[29],[30],[31], common hopping [32],[33],[34], and parallel-rendezvous [19],[15],[35],[36],[37],[38],[39]. Each approach has its specific advantages and weaknesses.

2.1.1 Dedicated Control Channel:

Dedicated control channel approaches as in Dynamic Channel Assignment (DCA)[6], splits the bandwidth into one control channel and N data channels. Each node is equipped with two radio interfaces, one operating on the control channel and other on any data channel. Nodes exchange control information on a fixed control channel to gain access to any data channel. Due to a common control channel, multi-channel hidden terminal problem simply does not exist in DCA. Synchronization is not needed for this approach. Nodes are aware of the neighbor activities all the time. Broadcasting information can be sent on the control channel. But as the density of nodes increases, the control channel exchanges also rapidly surmounts resulting in a bottle neck on the control channel. The upper bounds on number of data channels that a control channel can support needs to be considered [20], [21]. The dedicated control channel causes resource wastage of precious bandwidth, especially when only a few channels are available, as that in 802.11b. Two transceivers per host increases cost of a node and makes it expensive. The energy consumption of the node is nearly doubled. Having additional transceiver also increases the size of the sensor and makes it less practical in some cases.

2.1.2 Split Phase:

Split phase approaches are commonly based on global time synchronization. The time axis is divided into periodic contention phase and data transfer phase. During contention phase, all nodes listen to an agreed upon common channel and contend with each other to reserve the data channels. Nodes that had successful negotiations can involve in data transfer during data transfer phase. Other nodes which cannot obtain a channel will have to wait until the next contention phase. After data transfer is complete, ACK may also be sent on the same data channel. At the end of data phase, or upon the completion of DATA/ACK, whichever is earlier, nodes switch to the control channel for the next control phase, and the cycle repeats. The control channel is also reused during data transfer. This kind of multi-channel operation is mainly based on a global temporal synchronization among the nodes. Beacons or in some cases, GPS can be used to provide synchronization, so that all nodes know the start of control phase. Nodes can have some internal timers to demarcate the end of control phase and the start of data transfer phase. A big challenge arises as to which node or set of nodes will co-ordinate the sending of beacons. In infrastructure mode of operation, the access point takes care of sending beacons and other such centralized operations. However, to achieve time synchronization in an ad-hoc network is extremely difficult. In this approach, multi-channel hidden terminal problem is eliminated since nodes are cognizant of channel reservations during contention phase. Since a single transceiver is enough, there is no additional cost or energy spending. All channels are used during the data transfer phase, so bandwidth is reused unlike dedicated control channel approach. However temporal synchronization among the nodes is needed, as all nodes have to enter contention phase or data transfer phase at the same time. During contention phase, the common channel is congested by the control signaling traffic and can get saturated as the network size increase. At the same time, the data channels are unused during contention phase. Nodes which failed to have channel access rights in the contention phase need to defer until the next contention phase.

2.1.3 Common Hopping:

In Common Hopping, nodes follow a common hopping pattern and periodically switch between the different channels. Nodes which want to communicate negotiate with each other, suspend hopping and stay on the channel to engage in data transfer. Once the data transfer is complete, they re-join the hopping sequence. In single rendezvous, only a single agreement can happen between a sender and receiver on a given channel at any time. Hopping approach spreads the control signaling overhead across all the available channels. This approach needs only a single radio. Congestion on a particular channel is avoided. However, this approach require tight time synchronization as a must, which is very difficult to achieve. In addition, broadcasting is a real issue, since nodes are on different channels at various points in time. As channel hopping is frequent, channel switching delay is also a significant overhead.

2.1.4 Parallel Rendezvous:

In another category of Parallel Rendezvous, multiple handshakes can happen simultaneously on all the available channels. As such the control overhead is spread across all the channels. The support for increasing number of channels is good. When channels are numerous and packets are short, parallel rendezvous perform better than dedicated control channel approaches due to the elimination of control channel bottleneck [40]. However, a tight global synchronization is needed and support for broadcast is not adequate.

CHAPTER 3

OVERVIEW OF SELECTED MAC PROTOCOLS

3.1 Single channel Protocols

The single channel protocols based on 802.11: simple 802.11 and RTS-CTS option, are discussed in brief.

3.1.1 Basic Distributed Co-ordination Function (DCF)

802.11 provides simplified access mechanism for nodes to access the channel in a distributed manner. A node looking to send data senses the channel for a small time called DCF Inter Frame Space (DIFS) and if the channel is free, it can send DATA immediately. The receiver sends an acknowledgement frame ACK. In case, the channel is not free, the sender waits until the channel is available, and then waits a random backoff plus a duration of DIFS and send DATA. At any time if the channel becomes busy, the backoff timer is frozen and resumes once the channel is available. The back-off timer is a random number of slots from the contention window (CW). Some sort of fairness is achieved by employing randomness in back-off calculation. Setting the range of the CW to an optimum value is very important. If the window is too small, too many nodes will start transmission at the same time, and if the window size is too large, nodes have to experience a very long delay before the data transmission. The nodes double their CW when there is a collision or if a node does not respond to RTS. The operation of basic DCF is given in Figure 3.1.

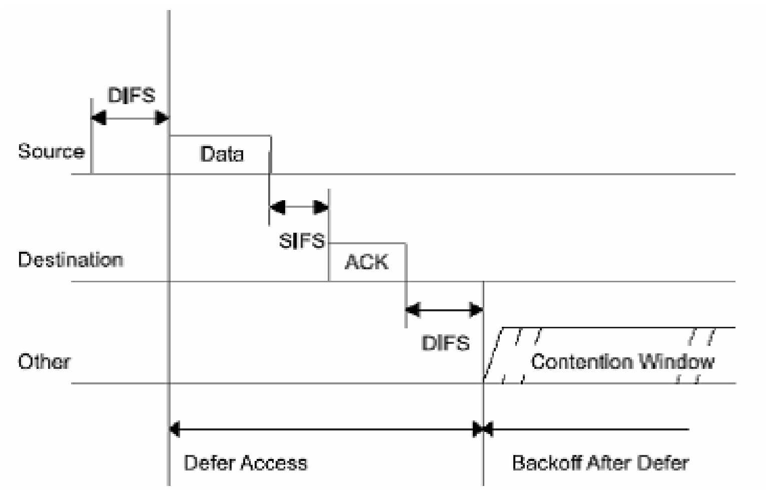


Figure 3.1: Basic Distributed Coordination Function as in [1] (Fig 2.)

3.1.2 DCF with RTS-CTS

Another 802.11 option uses four-way hand-shake RTS/CTS/DATA/ACK and CSMA/CA to reduce collision among the contending nodes. When a node has data to send, it senses the medium for idleness and sets the back off timer (0,CW).

At any point during the back-off countdown, if the medium becomes busy, the back-off timer is paused until the on-going transmission is over and then counts down from the same value, once the medium is available. On expiry of backoff timer, the sender waits for DIFS and if the medium is still idle, the node sends a Request-To-Send (RTS) packet to the receiver. The receiver, if it is available to receive data, will respond with a Clear-To-Send (CTS) packet. The sender receives CTS and starts sending DATA. On successful reception of data, the receiver sends ACK to the sender. RTS and CTS signals specify how much time it will occupy the medium, in a field called NAV (Network Allocation Vector). During transmission of RTS, the neighbor nodes of sender will get to know the NAV, and defer from any transmission for the time of NAV. Similarly, using CTS signal, the neighbor nodes of receiver update their NAV and stop doing anything until the time of NAV. The operation of RTS-CTS is illustrated in Figure 3.2. Short Inter Frame Space (SIFS) wait time is employed before sending CTS, DATA, and ACK. Any node looking for transmission needs to wait for a period of DIFS before sending. SIFS is smaller than DIFS, which means the node will get to know a transmission is underway and defers from sending. Thus, the four-way handshake of RTS-CTS-DATA-ACK effectively prevents collisions. Data frames can be exchanged without any collisions as those are now restricted to control frames.

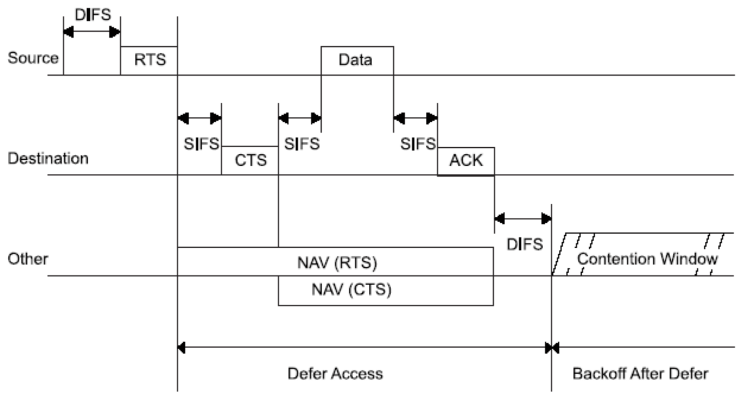


Figure 3.2: RTS/CTS mechanism in 802.11 as in [1] (Fig 3.)

3.2 Multi-channel Protocols

3.2.1 Multi-channel MAC (MMAC)

In MMAC, the time is divided by beacon interval with a negotiation window of 20ms and data transfer window of 80ms. In negotiation window, nodes with data send ATIM-REQ to the receivers and try to reserve a channel. Receiver select a common channel and respond with ATIM-RES upon which sender once again sends ATIM-ACK. During the data window, the agreed upon pair switch to the selected data channel and engage in 802.11 type of contention to get channel access.

3.2.2 Improved Contention Free Multi-channel MAC (ICMMAC)

In ICM-MAC [10], the negotiation phase is different. The negotiation phase is divided into a large number of timeslots. Nodes with data can only contend at the start of timeslots. Such nodes occupy a slot at random and send RTS. If RTS does not collide with another RTS, and if the receiver and a channel is available, CTS is sent. Sender acknowledges with an ATS (Acknowledge-to-Send) and wait for data transfer phase for sending the data. In case the sender does not get a response, it may choose to stay silent or send to another neighbor. This is however dependent on its probability of success dependent on the number of free channels at its side. If a node has one channel out of three total data channels free, it has 1/3 or 33% success probability. If a node has two out of three channels available, the success probability is 67% and so on. Also, this protocol applies an unconventional queueing policy. If the packet at the head of queue cannot be sent, it selects the next packet and tries to send it. Nodes in the network need to be modified to adapt to this uncommon queueing policy. Since negotiation phase is divided into a number of timeslots, the synchronization issue becomes even worse.

Typically the protocol uses a optimal value of 20 timeslots per negotiation window. Each timeslot is fixed as a sum of RTS +CTS+ACK apart from propagation delays. In DSSS, each RTS consumes 560 µs (44 bytes for RTS, 2 bytes for Channel Status, and 24 bytes for PHY header, sent at basic rate of 1 Mbps), CTS needs 436 µs (38 bytes default, 2 bytes for Channel Status, and 24 bytes for PHY header) and again ATS needs a similar 436 µs as that of CTS. So the overall overhead of RTS, CTS, and ATS for a single time slot is 1432 µs or 1.432 ms. For 20 time slots, a window of 28.64 ms is needed. The protocol sends a single fixed length data frame during data transfer. Considering the maximum data frame of 2304 bytes sent at typical 2Mbps in DSSS, the total time for data transfer is

9.216 us. Here, the control window is three times longer than the data window. As this earlier scenario exposes that a lot of time could be wasted in sending the control signals. The unconventional queuing policy and the stringent need for time synchronization makes it less compatible with existing 802.11 nodes.

3.2.3 Dynamic Channel Assignment (DCA)

Dynamic Channel Assignment [6] is a multi-channel MAC protocol using two transceivers: control transceiver and a data transceiver. The control transceiver always operates on the channel reserved exclusively for control signaling (control channel), and the data transceiver operates on one of the data channels for data exchange.

By utilizing more than one channel, the protocol results in higher throughput and lower delay than 802.11. With a dedicated radio and exclusive channel for control signaling, nodes always have a full visibility of the channels usage. Multi-channel hidden terminal problem and deafness problem is simply absent due to this feature. While nodes are busy in exchanging data on the data channel using the data transceiver, the control transceiver will hear the negotiations on control channel and update the channels status. When data transceiver of node is involved in data exchange, the control transceiver will not respond to any incoming RTS. But this issue of missing receiver is less serious. Further in DCA, there is no need for temporal synchronization among the nodes and broadcast messages can be sent on the control channel.

While the extra transceiver and an exclusive control channel is able to eliminate all problems inherent in a multi-channel environment, it also result in high cost, more energy spending, and increase in node size. Depending on the application, all of these are deterrent factors. For example, in dense deployment of nodes, the size of nodes need to be small. Also, energy conservation is a very important factor. High energy consuming nodes are not suitable. Expensive nodes are not preferable since hundreds or thousands of nodes might be deployed. Besides these factors, the exclusive control channel incurs a huge penalty in terms of bandwidth scarcity. For example, 802.11b has only 3 orthogonal channels. A dedicated control channel consumes about one-third (33%) of bandwidth leaving only two-third (67%) for data transfer.

The contention mechanism on the control channel in DCA follows the same as that of 802.11. Figure 3.3 illustrates typical operation using DCA. When a node has data, it sends RTS to the destination with the available list of free channels. The destination node checks its list of available channels and chooses one and sends CTS to the sender. The sender and receiver immediately switches to the agreed data channel and use their data transceivers for the data exchange. The sender also sends reserved (RES) signal on the control channel to announce the selected channel and the channel usage time. When a RTS is received, the neighbors update their NAV until the end of RTS-CTS-RES and an additional DIFS. Then the nodes with data resume their backoff decrement counter and contend for transmission. While a data transfer on a channel is on-going, the control negotiations for that channel can start at a period of RTS+CTS+DIFS before the channel release time. This is by virtue of a dedicated transceiver on the control channel, and that the control and data transceivers can work in parallel.

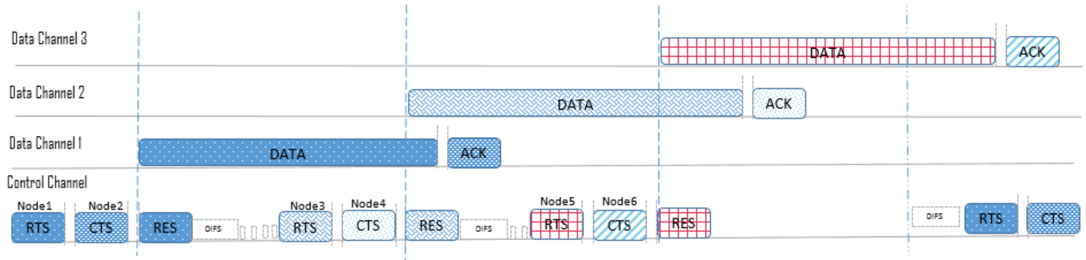


Figure 3.3: Dynamic Channel Assignment

3.2.4 Bidirectional Multi-Channel MAC (BiMMAC)

The four-way handshake (RTS-CTS-Data-Ack) negotiation in 802.11 can avoid data collisions, but the time spent in backoff, contention and RTS, CTS signals creates an overhead for each frame and effectively reduces the network throughput. BiMMAC [9] seeks to overcome this control overhead by a novel approach. Once a pair of nodes agrees on a data transfer, the receiver can send an extra frame without the cost of these contention and backoffs. When a node has to respond to a RTS, it checks if it has any data to the sender. If so, it adds that time to CTS. When the first data frame from sender (A) to receiver (B) is successful, node B send the second frame to node A. Thus the node A knows that the transfer of first frame is successful. i.e, the transfer of second frame from node B is a piggybacked acknowledgement to node A. A then sends ACK to B to confirm the receipt of second frame. In the absence of a data packet to sender, an ordinary ACK packet is sent from receiver. An illustration of this is depicted in Figure 3.4. Thus the second frame from node B effectively skips all the contention, waiting, back-off, and RTS-CTS agreement. A visible downside of this approach is that other nodes suffer a large delay in acquiring the channel due to transfer of two time frames in a single negotiation. Another side effect is that even the sender needs to wait for an additional frame time before the next data exchange.

The channel selection mechanism is similar to that of DCA. When a node has data to send, it sends RTS on the control channel with the list of available channels on its side. The receiver, on getting the RTS checks its list of free channels and selects a commonly available channel. If no common channel is available, or if the node is busy in data exchange, the RTS request is ignored. If there is some data intended for the sender, the receiver node updates the channel release time by adding the data transfer time in its CTS frame. The receiver then switches to the data channel and get ready to receive data. The sender on receiving the CTS, announces the selected data channel and channel release time to its neighbors in another frame called Channel Reservation (CRN). Then it switches to the data channel and starts sending data. The neighboring nodes which hear the control exchange freeze their back-off timers and defer from contention for the duration of transmission.

BiMMAC does not need temporal synchronization among the nodes, which is nice, since achieving a global time synchronization is practically very difficult. The protocol uses a single transceiver and listens to a commonly assigned control

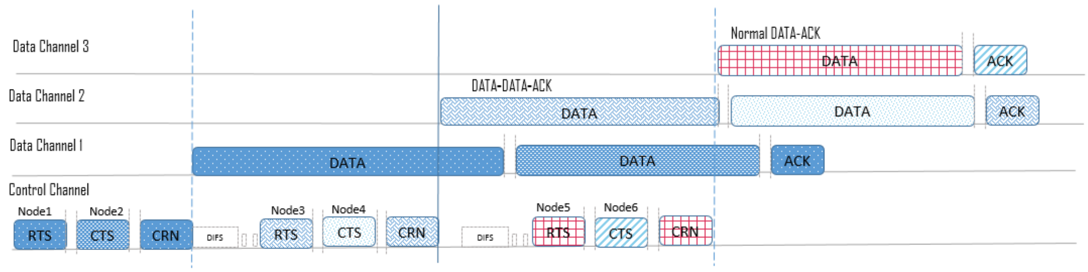


Figure 3.4: Bidirectional Multi-Channel MAC

channel by default. The single radio in BiMMAC as opposed to two radios in DCA, translates into good cost savings, energy conservation, and reduced sensor size. However, like DCA, it uses a commonly agreed channel for control signalling. The penalty it pays for using a single radio is that it is prone to multi-channel hidden terminal and deafness problems. At any given time, a half-duplex transceiver can only either send or receive on only one channel. As such, in BiMMAC, nodes in data transfer are deaf to the exchanges on the control channel and may select a busy channel inadvertently in the next frame exchange, which cannot be eliminated completely. By selecting a previously used channel, the data collisions due to the deafness problem can be solved to a reasonable extent. BiMMAC has comparable performance to DCA. Nonetheless, DCA and BiMMAC are hampered by scarce of bandwidth due to a dedicated control channel.

3.2.5 Asynchronous Multi-Channel MAC (AMMAC)

AMMAC [41] differs from DCA, BiMMAC by reusing the control channel. It eliminates the multi-channel deafness problem, by using a single radio, and an asynchronous mode of operation with certain conditions which are listed below.

- Nodes receiving a control frame (RTS,CTS,etc..) needs to act upon it, irrespective of whether the frame is received correctly or in error.

- Then, nodes that completed a data transfer observe a mandatory period greater than the maximum data frame. The receiver will a kind of duplicate CTS signal called Announce-To-Send (ATS) immediately after CTS.

Due to hidden terminal in RTS/CTS, two nodes sending RTS at the same time causing a collision at the third node or atleast the one node is deaf to the other RTS. It is also possible that a when node sends CTS, a neighbor may send RTS simultaneously. By employing another ATS signal, the neighbor nodes who missed RTS or CTS, get another chance by means of ATS. The sender and the receiver announce their intention twice: sender (RTS, ATS) and receiver (CTS, ATS). Since nodes have to act upon error RTS or CTS frames, the nodes get to know that another transmission is going on and backs off. The hidden terminal in RTS/CTS is thus resolved.

In AMMAC, the channel selection and data transfer happen on a per packet basis. The deafness issue in multi-channels is addressed by another means. A node returning from data transfer waits in the control channel for a minimum of a time greater than the maximum data frame. By this approach, a node that just-completed data transfer (let's say node A) and returning to the control channel and another node which obtained the data channel at the same time (let's say Z) can get to know each other. Before node A starts another RTS, node Z will be back to the control channel and will be ready to hear the control exchanges.

This additional waiting time, although an overhead, serve to eliminate the multi-channel hidden terminal and can possibly give a collision free operation. The lack of need for a temporal synchronization is another good feature.

The contention mechanism follows 802.11 on the control channel like DCA and BiMMAC. Nodes will send RTS, CTS, and ATS on the control channel and switch to the agreed data channel for data transfer. The operation of AMMAC is illustrated in Figure 3.5.

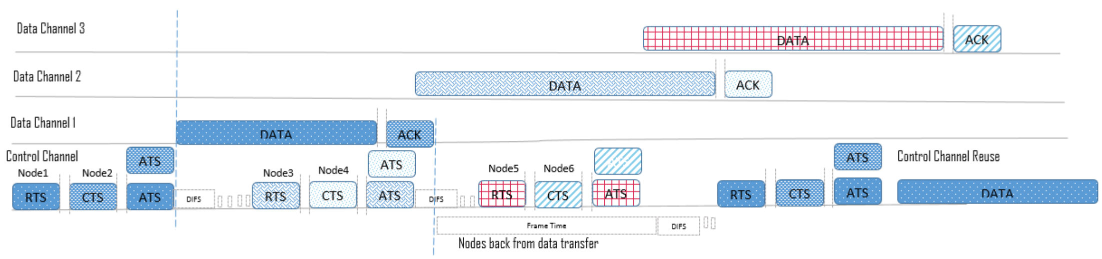


Figure 3.5: Asynchronous Multi-Channel MAC

CHAPTER 4

PROPOSED ASYNCHRONOUS BIDIRECTIONAL MULTI-CHANNEL MAC

The downside in AMMAC is that the nodes returning from data transfer are penalized by a waiting of a maximum data frame time. Moreover, the control channel is also reused. When control channel is used for data transmission, other nodes with data have to wait for control negotiations until the completion of ongoing transmission, though at the same time, the data channels are empty and available for use. In a sense, the nodes which used data channels are penalized once by a mandatory waiting of maximum data frame time on control channel, an additional data frame time in case of any transfer on control channel, plus other contention and backoffs.

By using two frame transmissions like BiMMAC on data channels and a maximum single data frame on control channel, the advantages of AMMAC and BiMMAC can be reaped. So the nodes can reuse the control channel and at the same time the receiver can send an extra frame on data channels skipping the control negotiations. The enhanced asynchronous bidirectional multi-channel MAC (ABMMAC) will thus have operation mechanisms overlapping AMMAC and BiMMAC.

4.1 Working of ABMMAC

ABMMAC transmits data on a per-packet basis and uses most of the features of AMMAC like contention mechanism on the control channel using RTS/CTS/ATS, control-channel reuse, one frame waiting period after data exchange, etc., On data channels, the data transfer follows a BiMMAC type of operation. Nodes have N data channels and one control channel, which can be used for data exchange. By default, all nodes listen to a common channel marked for control signaling. Nodes with data, first send RTS to the intended receiver on the control channel. RTS includes the list of free channels available at the sender side. Neighbors hearing the RTS need to defer for an initial time of RTS+SIFS+ATS. The receiver on receiving RTS, checks its list of free channels and selects a common available channel. If more than one channel are available, a channel is chosen randomly and CTS is sent on the control channel. Neighbors receiving CTS defer until the time needed to complete data transfer. After a small time SIFS, the sender and receiver send ATS in parallel, informing the neighbors of the selected channel and the channel occupation time. Neighbors update their defer time until the end of channel occupation. Having sent ATS, the sender and the receiver switch to the selected data channel to proceed with data transfer. ACK is sent on the data channel itself. When the process is complete, the sender-receiver switch to the control channel and take cognizant of the control messages exchanged. They need to wait for a minimum data frame time before they start the next transmission. The illustration of this is given in Figure 4.1.

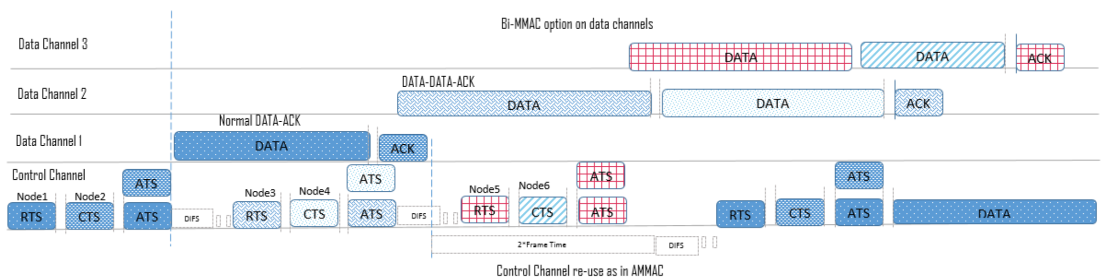


Figure 4.1: Asynchronous Bidirectional Multi-Channel MAC

4.2 Comparison of features of DCA, BiMMAC, AMMAC, and ABMMAC:

The similar features of DCA, BiMMAC, AMMAC, and ABMMAC is shown in Table 4.1 and comparison of other features is given in Table 4.2

Table 4.1: Similar features of the selected multi-channel protocols

Common Features of DCA, BiMMAC, AMMAC, and ABMMAC
A common channel agreed for control signalling
No Temporal Synchronization Needed
Support for Broadcast
Moderate Channel Switching Delay
Missing Receiver Problem Present
Distributed Random Access
Contention mechanism similar to 802.11

Table 4.2: Comparison of features of DCA, BiMMAC, AMMAC, and ABMMAC

	DCA	BiMMAC	AMMAC	ABMMAC
Issue of Multi-channel Hidden Terminal	No	Possible	No	Possible
Control Channel overhead	Less	Less	High	High
More than one data frame in a handshake	No	Yes	No	Yes
Number of Transceivers	2	1	1	1
Exclusive Control Channel needed	Yes	Yes	No	No
Channel Selection*	Random	Last Used	Random	Last Used
Cost of Node	High	Less	Less	Less
Size of Node	Big	Small	Small	Small
Battery Consumption of Node	High	Low	Low	Low
Scarcity of Bandwidth for data channels	Significant	Significant	Less Significant	Less Significant

* Subject to availability of common channels

CHAPTER 5

SIMULATION SETUP AND PERFORMANCE METRICS FORMULATION

5.1 Selecting protocols for simulation

We chose multi-channel protocols those do not require temporal synchronization among the nodes. Considering the difficulties associated with global synchronization, split phase protocols such as MMAC, ICMMAC, and channel hopping MAC approaches are ignored. We consider those protocols that are asynchronous and based on 802.11 contention mechanism. DCA, AMMAC, and BiMMAC fit our category. The chosen protocols are given below:

- **Single Channel:**

    a) 802.11 basic DCF

b) 802.11 with RTS-CTS

- **Multi-Channel:**

  a) DCA

  b) BiMMAC

  c) AMMAC

  d) ABMMAC

DCA uses two transceivers while DCA, BiMMAC, and AMMAC use a single half duplex transceiver. ABMMAC is our proposed protocol by incorporating certain features of other protocols.

5.2 Commonly Applied Performance Metrics

Choosing design parameters for MAC in wireless networks is application specific. A MAC suitable for particular environment may not suit the other. We discuss the commonly applied performance characteristics first before choosing the metrics for the simulation.

- **Throughput:** The throughput indicates the amount of data transmitted from source to destination in a unit of time, measured in bits per second. The throughput is derived by calculating the data packets received at the nodes divided by the total simulation time. Ideally, total throughput of a multi channel (N channels) protocol is N times the throughput of a single channel protocol, considering N transceivers. However protocols using a single transceiver may experience channel switching delays and overhead channel negotiations thus impacting the throughput. For throughput, the higher the value the better. The channel access mechanism should be designed to yield high data throughput.

- **Average Packet Delay:** Packet delay is the time since a packet comes to the link layer of the sender till the time it reaches the same layer on destination. The channel negotiation delays, transmission delays, and queueing delays all form part of the packet delay. A packet may pass through multiple hops to reach the destination. The basic concepts concerning delay are detailed further below.
  - **Processing delay:** A time it takes for a node to read the packet header and decide if the packet is destined for the node or it has to be forwarded.
  - **Queueing delay:** When a packet arrives at a node and no other packet is waiting to be transmitted, it can be transmitted immediately. However, if there are other packets pending, the new arrival is buffered into a queue and kept in waiting, until all the packets before are transmitted successfully or dropped from failed re-transmissions. High traffic conditions cause packets to collide, increase contention window and result in data re-transmissions, all of which translates to greater queueing delay.
  - **Transmission delay:** Transmission delay is a function of packet

length and the rate of transmission and denotes the time it takes to push a packet into a channel. If the length of the packet is L bits and the rate of transmission is R bps, then the transmission delay can be given by L/R seconds.

  - **Propagation delay:** Propagation delay indicates the time needed for a packet to reach from the sender to the destination. The factor is a function of distance and the medium in which the bits travel. If the distance between the source and destination is 'd' meters and the propagation speed is 's' meters per second, then the propagation delay can be calculated as d/s. In wireless, the propagation medium is air and generally equals to the speed of light ($3 \times 10^8 m/s$).

  Processing and Propagation delay are generally in the range of few microseconds and negligible.

- **Loss Factors:** The bit error rate, collision rate, packet drop rate all lead to packet loss. For these metrics, lower is better.

- **Hidden/Exposed terminal problem:** Hidden terminal problem cause collisions while exposed terminal problem denies a potential sender from data exchange, both of which reduces the network throughput.

- **Energy conservation:** Energy conservation may be important as nodes may be operating on battery power sources. Reducing the idle listening times, collisions, overhead, etc., can serve to save the energy [42].

- **Fairness:** This is a measure of the extent to which the protocol allocates spectrum bands equally, or on a weighted basis, to all nodes. Fairness can be considered in a strict sense or loose sense and needs to be designed correctly. For example, certain nodes may not have any data to transmit and stay idle all the time. Such nodes cannot be included for fairness calculations.

- **QoS support:** This metric indicates if the MAC can support real-time traffic such as audio streaming, video, or other priority to some time constrained data flows.

- **Packet overhead:** The overheads include per packet overhead (headers, trailers), and exchange of control packets. For example, a multi-channel MAC operating with 11 channels, as in 802.11a, can choose to send channel status of all the channels. If each of the status consumes 1 byte, 11 bytes may be increased on RTS, CTS, and other control signals. Higher overheads translate into more transmission times and good strategies can reduce such overheads. The overhead decreases the network throughput especially when the packet size is very small (typically less than 2Kb) [43], otherwise the impact of the overheads is minimal [44].

5.3 Simulation Setup

We developed a flexible simulation tool in python to compare the single and multi-channel protocols. The correctness and reliability of the simulator is detailed in the next chapter.

5.3.1 Assumptions

MAC layer is decoupled from PHY and routing layers so as to study the effectiveness of the MAC protocol. The multi-channel protocols implemented are strictly logical extensions of 802.11 and as such assumed to have 802.11 like behaviour in other layers of TCP/IP stack. The assumptions used in our simulation are totally derived from previous works [1], [43], [45], [46], [47], [41]. The assumptions are listed below:

- Saturated traffic conditions is assumed, which effectively means that nodes will have a frame to transmit all the time.
- The channel is error free. Frames are lost only due to collisions.
- The nodes are in transmission range of each other in a single collision domain. The category of multi-channel hidden terminal problem is considered.
- The channels used are orthogonal to one another. Transmissions in one channel will not interfere with any transmission on other channels.
- Packets arrive at nodes according to Poisson process. A packet arrival at a node is independent of the other arrivals.
- There is no distinguishing among the packets. All packets have equal priority.

- Data frames have a constant fixed size.
- Frame exchanged are: RTS, CTS, ATS, DATA, and ACK. No other frames are considered.
- All nodes use the same PHY layer using Direct Sequence Spread Spectrum (DSSS) at a rate of 1 Mbps.
- All nodes transmit with the same data rate. Varying data rates at various nodes at the same time is not considered.
- Both data and control frames are sent at the rate of 1 Mbps, unless specified otherwise.
- In case of collision, the frames are discarded. The capture effect is not considered.
- All simulations are run for time of 10000 data frames. The chosen frames count is validated and found to provide sufficient accuracy in populating the performance measures.

5.3.2 Performance Metrics

For the performance characterisation of the single and multi-channel protocols, most of our metrics are similar to that of [1]. The metrics are listed below:

- **Aggregate Throughput**: Aggregate throughput denotes the total data sent in in the given simulation time measured in Mbps. The higher the throughput the better it is.

- **Packets Received**: This measure is kept in addition to aggregate throughput and denotes the total packets received successfully.

- **Throughput efficiency**: Throughput efficiency gives a measure how much of the channel capacity is used. The value ranges from 0 to 1.

- **Delays**: We consider mainly average queueing delay and packet delay. Packet delay is calculated by summing up queueing delay and transmission time. The rest of delays are negligible and ignored. The delays of all the nodes is calculated and an average delay is obtained for both the metric.

- **Frame Drop Ratio (FDR)**: This indicates the ratio of frames dropped to the total frames sent. The lower the FDR the better it is.

- **Jain's Fairness Index (JFI):** Jain's fairness index [48] describes how similar and fair the bandwidth or a channel is allocated to each node. JFI is calculated by

  $$\text{JFI} = \frac{|\sum_{i=1}^{N} X_i|^2}{N \sum_{i=1}^{N} X_i^2}$$

  where N indicates the number of nodes and X_i is the throughput for the i^{th} node.

5.3.3 Simulation Parameters

The physical layer parameters considered for simulation are 802.11b specific. The parameters common to all MAC protocols are given in Table 5.1, and the parameters for single channel 802.11 with RTS-CTS are listed in Table 5.2.

Table 5.1: Common Parameters

Parameter	Value
CWmin(slots)	31
CWmax(slots)	1023
SlotTime	20 µs
SIFS	10 µs
DIFS	50 µs
Tack	304 µs
Channel bit rate	1 Mbps
DATA Frame	8224 bits
MAC Header	224 bits
PHY Header	192 bits
ACK	112 bits + PHY Header
Retry Limit	7

Table 5.2: Parameters for 802.11 RTS-CTS

Parameter	Value
RTS	160 bits + PHY Header
CTS	112 bits + PHY Header
ACK	112 bits + PHY Header

Table 5.3: Parameters for multi-channel protocols

Parameter	Value
Control Channel	1
Data Channels	2
RTS	168 bits + PHY Header
CTS	120 bits + PHY Header
ATS, RES, CRN	120 bits + PHY Header

The parameters applicable for multi channel protocols are given in Table 5.3. In multi-channel protocols, after a successful RTS-CTS exchange, the sender typically sends another control frame to neighbors to announce the channel selection. The control frame has the same functionality across all the protocols but the naming differs. AMMAC uses ATS frame, while DCA and BiMMAC use RES and CRN frames.

CHAPTER 6

SIMULATOR VALIDATION

6.1 Comparison of Simulator with existing literature

DCA, AMMAC, BiMMAC, and ABMMAC are logical extensions of 802.11. Their underlying contention and back off mechanism is the same as that of 802.11. As such, ensuring the accuracy of these basic DCF mechanism and RTS-CTS is very important. We took extensive efforts to verify the correctness of our simulator. The results of 802.11 basic Distributed Co-ordination Function (DCF) and RTS-CTS from our simulator are compared with previous works such as Al-Akeel et al [1], Haiatao Wu et al [47], Bianchi et al [45] [46] [49], and Chatzimisios et al [43]. These are seminal works in performance analysis of 802.11 and highly cited in literature.

6.1.1 Comparison of access mechanism with Wu's

The throughput of 802.11 basic access mechanism with respect to the number of nodes from our simulation is shown in Figure 6.1.

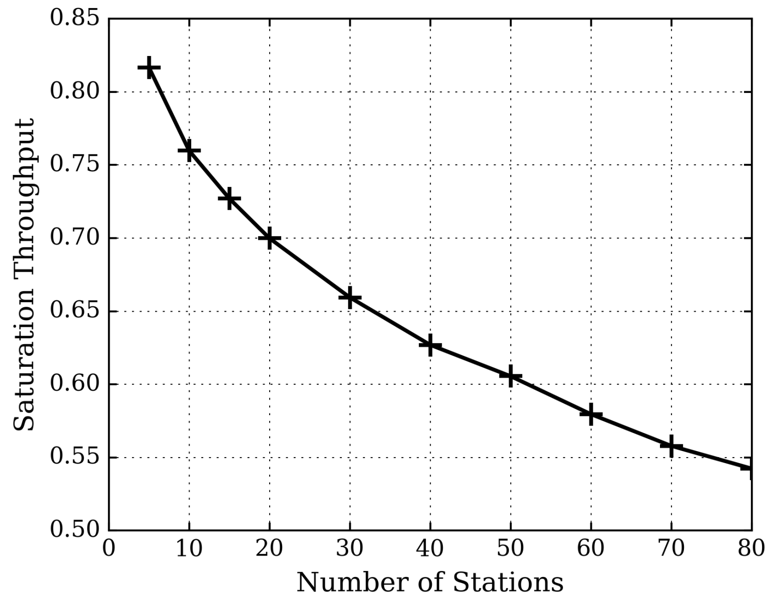


Figure 6.1: Comparison of 802.11 basic access Throughput with Wu's

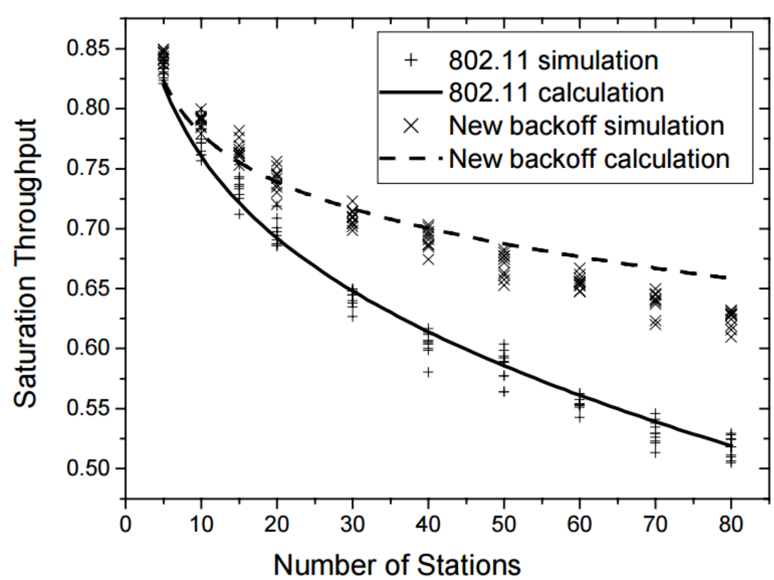


Figure 6.2: 802.11 basic access Throughput of Wu

As the number of nodes increases, the throughput degrades. Figure 6.1 is compared against that of Haitao Wu et al. [47] Figure 6.2 and the curve is found to be matching.

6.1.2 Comparison with Bianchi's

Bianchi et al. is one of the important works in throughput and delay analysis of 802.11. We compared our simulator against that of Bianchi. The parameters for this scenario are similar to that of authors'; packet payload 8184 bits, slot time 50 µs, SIFS 28 µs, DIFS 128 µs, ACK timeout 300 µs, MAC header 272 bits, PHY header 128 bits, ACK 112 bits+PHY header, RTS 160 bits+PHY header, CTS 112 bits+PHY header. The scenario in Figure 6.3 plots the saturation throughput for various configurations of initial contention window (W) and maximum number of stages (m), where m is the factor used for increase the contention window in case of collision. The initial contention window W=32 indicates that nodes initially chose a random value from this contention window. The stages m=3 indicates that on successive collisions, the window size is doubled, and this increase continues up to 3 stages, and no further. So, in this case, the utmost size of contention window is 256. After this, that window remains constant. Likewise, m value of 5 indicates that the window is doubled up to 5 stages i.e, 1024. The configurations for basic DCF mechanism are W=32; m=3, W=32; m=5 and W=128; m=3. For RTS-CTS, the configurations are W=32;m=3 and W=128;m=3. This comparison can help us to test the basic DCF simulation as well as that of RTS-CTS. The results show that the saturation throughput is higher when RTS-CTS is used. The results of basic DCF and RTS-CTS match Bianchi's Figure 6.4 output.

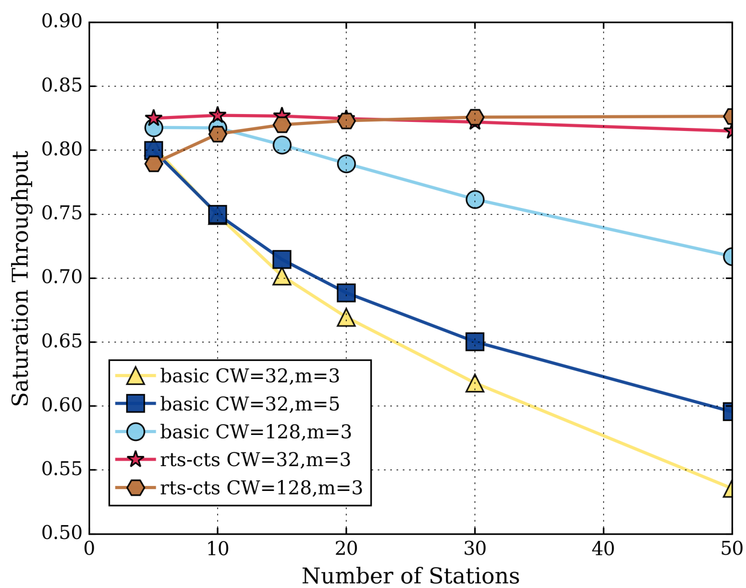


Figure 6.3: Comparison of Saturation Throughput with Bianchi's

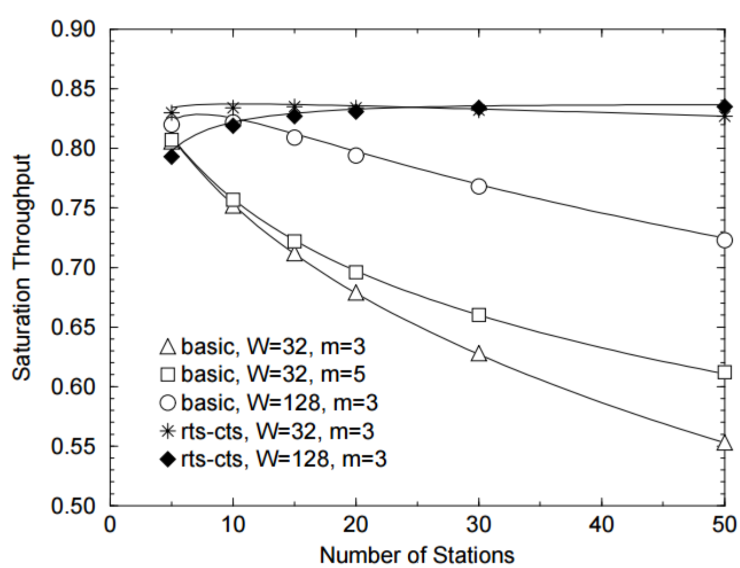


Figure 6.4: Saturation Throughput of Bianchi

6.1.3 Comparison with Chatzimisios's

Chatzimisios et al. [43] is another important in throughput and delay analysis of 802.11 protocol. We compare our simulator to this work using DSSS physical layer as in 802.11b.

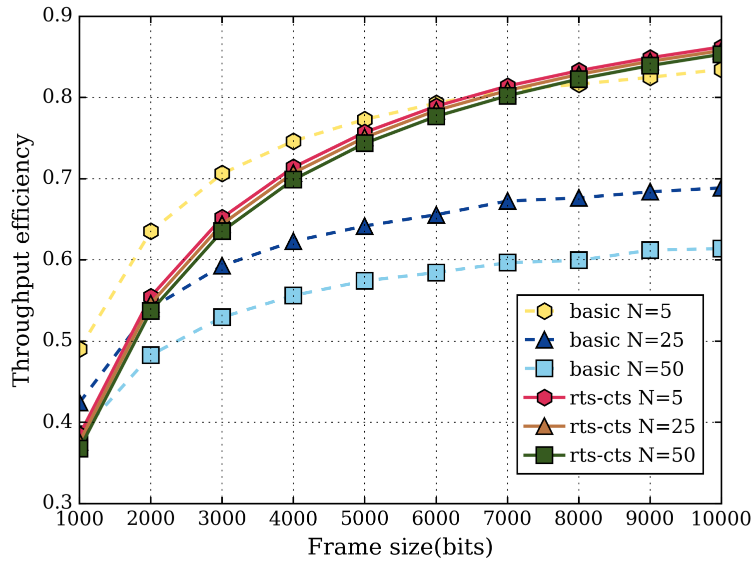


Figure 6.5: Comparison of Saturation Throughput with Chatzimisios's

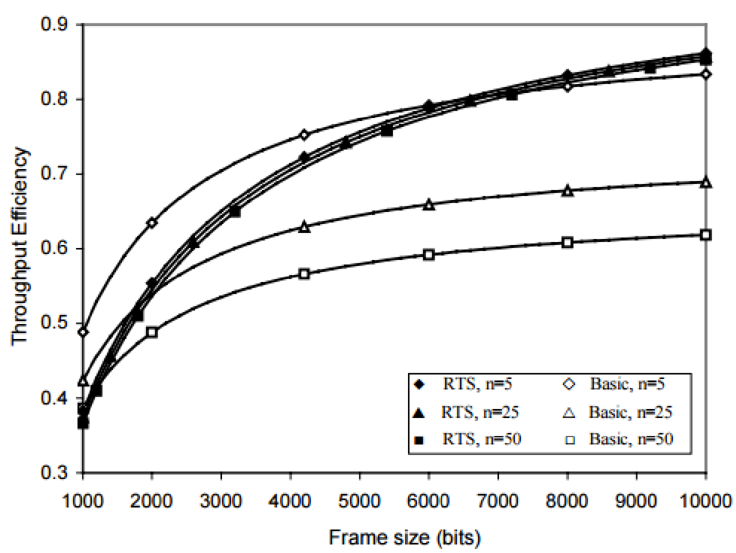


Figure 6.6: Throughput vs Frame Size for Channel rate 1 Mbps in Chatzimisios

The parameters are extracted from that of authors'; packet payload 8224 bits, slot time 20 µs, SIFS 10 µs, DIFS 50 µs, MAC header 224 bits, PHY header 192 bits, ACK 112 bits+PHY header, RTS 160 bits+PHY header, CTS 112 bits+PHY header. In this case, the frame sizes are varied from 1000 to 10000 bits and basic DCF and RTS-CTS mechanism throughput efficiency is compared for different number of nodes n=5,25,50. The results show that RTS-CTS mechanism is not efficient for smaller networks using smaller frames. However, the performance improves as the frame size increases and for higher number of nodes. Again, these results match closely that of Chatzimisios et al., Figures 6.5 and 6.6.

6.1.4 Comparison of Queueing Delay with Al-Akeel's

In order to verify the queueing delay, probability of frame drop, and Jain Fairness Index, we compare our work against Al-Akeel et al [1].

The saturation queueing delay for basic DCF for increasing number of nodes is shown in Figure 6.7. The authors graph for the same set of scenario is shown in Figure 6.8. As nodes increases to 120, the queueing delay increases up to 1200 ms in both the cases. The average packet delay also gives a similar trend and found to resemble Al-Akeel et al. output.

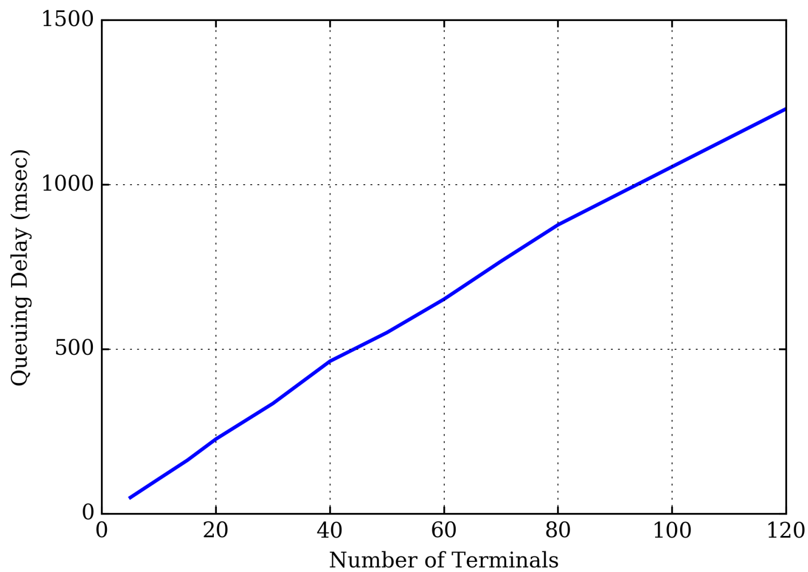


Figure 6.7: Comparison of Saturation Queueing Delay with Al-Akeel's

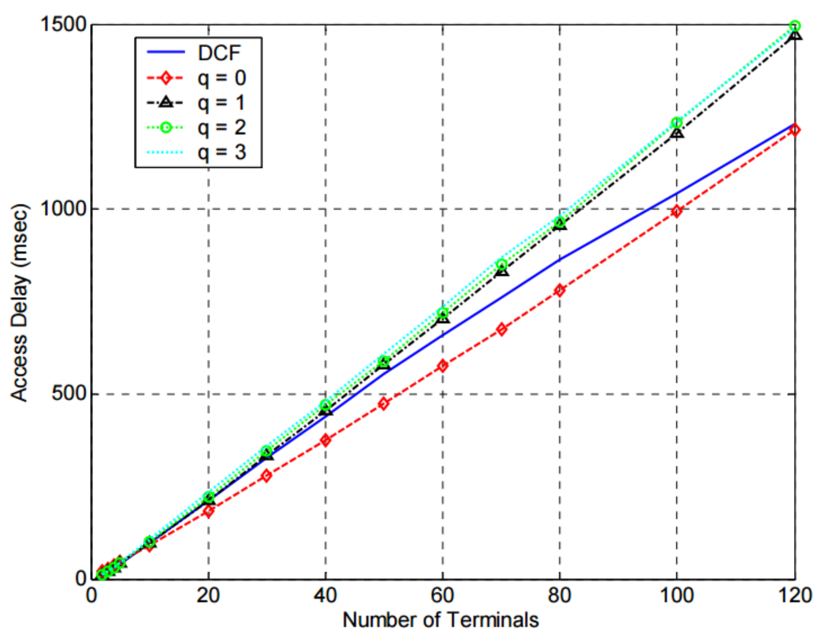


Figure 6.8: Saturation Queueing Delay of basic DCF as in Al-Akeel

6.1.5 Comparison of Frame Drop Ratio with Al-Akeel's

The frame drop ratio versus the number of stations for basic DCF is shown in Figure 6.9.

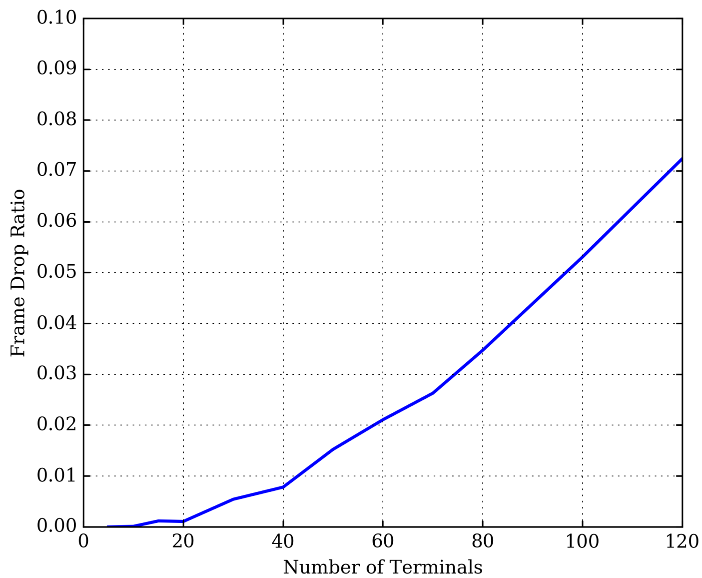


Figure 6.9: Comparison of Frame Drop Ratio with Al-Akeel's

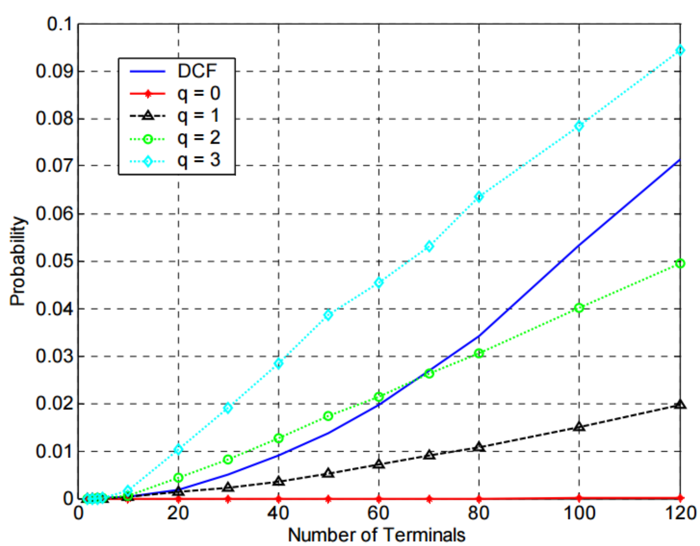


Figure 6.10: Frame Drop Ratio as in Al-Akeel

As the number of nodes grow to 120, the drop ratio increases up to 7%. Figure 6.10 in author also reported the same trend.

6.1.6 Comparison of Jain Fairness Index with Al-Akeel's

Again for JFI, the parameters are set similar to that of previous work, for set of 5 nodes and a window size m=1 normalized to the number of stations. The results are shown in Figure 6.11. The referred result is shown in Figure 6.12. As we can observe, the fairness index trend matches exactly with the mentioned work. From the results provided above, we can substantially conclude the accuracy of our 802.11 basic DCF and RTS-CTS simulation, and by extension, the accuracy of underlying contention mechanism of the used multi-channel protocols: DCA, AMMAC, and BiMMAC.

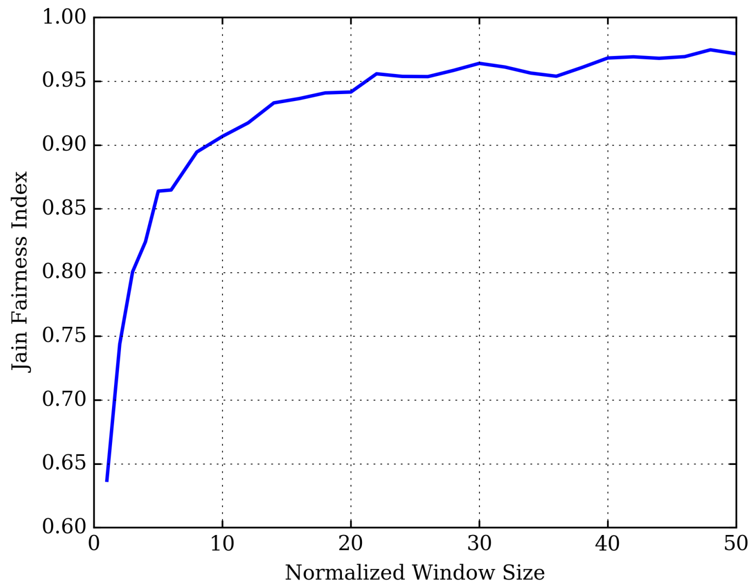


Figure 6.11: Comparison of Jain Fairness Index with Al-Akeel's

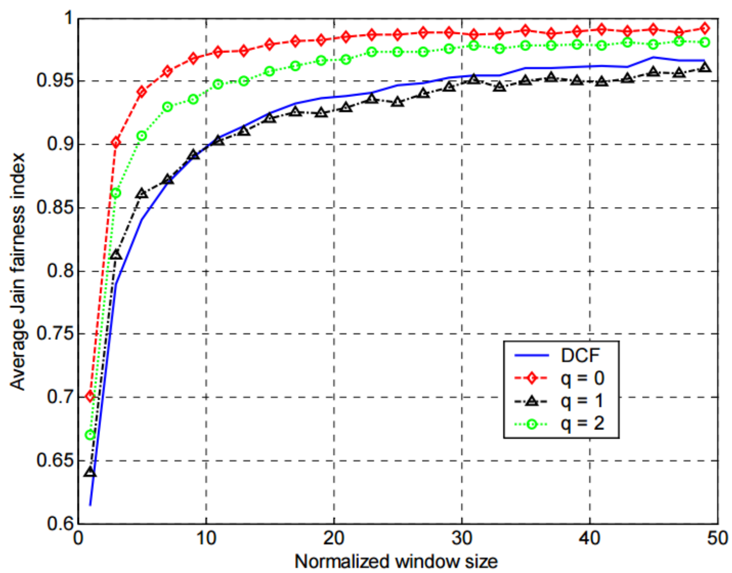


Figure 6.12: Jain Fairness Index as in Al-Akeel

6.2 Test for Maximum Aggregate Throughput

Table 6.1 shows the aggregate throughput of 100 nodes using 1 control channel and 2 data channels. The maximum aggregate throughput for single channel protocols is upper bounded by 1 Mbps, since they have jt one channel to operate all the time. The aggregate throughput of basic DCF (0.5118 Mbps) and RTS-CTS (0.8177 Mbps) are within the limits.

Table 6.1: Test for Aggregate Throughput

	Aggregate Throughput(Mbps)
Basic DCF	0.5118
RTS-CTS	0.8177
DCA	1.8451
BiMMAC	1.7686
AMMAC	1.9663
ABMMAC	2.1268

As for DCA and BiMMAC, they have two data channels to operate, and the aggregate throughput is bounded by a maximum of 2 Mbps. DCA have an aggregate throughput of 1.8451 Mbps and BiMMAC have 1.7686 Mbps, and both are below the maximum limit. As for AMMAC and ABMMAC, since they can ree the control channels, they have a total of 3 channels to operate. Hence, their aggregate throughput can reach a maximum of 3 Mbps. The aggregate throughput observed for AMMAC (1.9663 Mbps) and ABMMAC (2.1268 Mbps) conform to the limits.

6.3 Validations on BiMMAC, DCA, ABMMAC

As for comparison of multi-channels, the throughput gains of BiMMAC and DCA are close to each other. This is similar to the author observations in [9]. BiMMAC gives some bias to the successful nodes by allowing receiver to send additional frame. The factor of JFI th decreases, though less as compared to single channels. ABMMAC employ similar approach on data channels. We can observe that the Jain Fairness Index of both protocols are similar, as they have some similar channel grant policies.

By comparing our simulation results with existing literature, we substantiate that the custom made simulator is accurate and produces the right results.

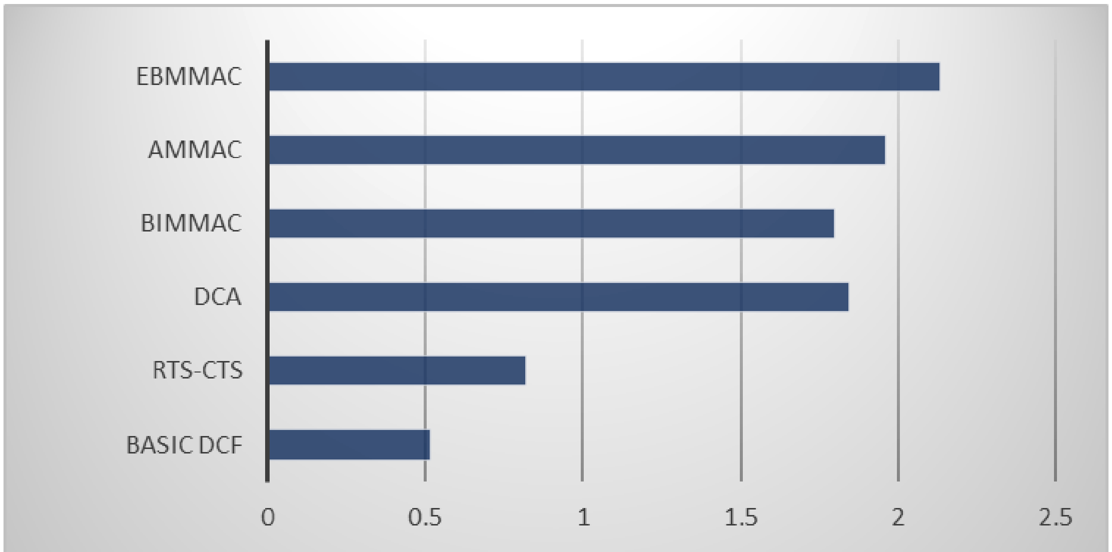
CHAPTER 7

RESULTS AND DISCUSSION

The performance of the protocols is studied for various network sizes. The metrics such as aggregate throughput, total packets sent, average queueing delay, average packet delay, frame drop ratio, and Jain Fairness Index are measured for various network sizes (20, 50, 100, 500) to mark small to large size networks. For multi-channel protocols, one control channel and two data channels are used. The simulation results are discussed below.

7.1 Aggregate Throughput

Figure 7.1 shows aggregate throughput for all the protocols for a network of 100 wireless nodes. Single channel protocols such as basic DCF, and RTS-CTS obtain an aggregate throughput of 0.5162 Mbps and 0.8176 Mbps respectively. In basic DCF, the increased contention of nodes leads to collisions of data frames. As network size grows to 100 nodes, the aggregate throughput in DCF already degraded to 0.5162 Mbps. RTS-CTS limits the collisions to control frames, and shows a better performance (0.8176 Mbps), 58.39% more throughput, than basic DCF.



	basic DCF	RTS-CTS	DCA	BiMMAC	AMMAC	EBMMAC
AggrThro (Mbps)	0.5162	0.8176	1.8453	1.8	1.9621	2.1331

Figure 7.1: Aggregate Throughput (AggrThro) Comparison for 100 nodes

With respect to multi channels, DCA and BiMMAC achieve an aggregate throughput of 1.845 Mbps and 1.8 Mbps respectively. The maximum throughput for DCA and BiMMAC is 2 Mbps, since they use two data channels. The control signalling are shifted entirely to a separate channel. DCA is 257.47% better than basic DCF and 125% better than RTS-CTS. DCA achieves this due to use of more than one data channel. BiMMAC has similar but slightly less throughput than DCA. But it achieves this performance by having one transceiver less. AMMAC gains 6.32% more throughput than DCA and 9% more throughput than BiMMAC, due to its reuse of control channel. ABMMAC performs better than all the other protocols. It has gains of 313.23% over basic DCF, 160.898% over RTS-CTS, 15.596% over DCA, 18.5% over BiMMAC and 8.7% over AMMAC. ABMMAC achieves these by virtue of multiple channels and reuse of the control channel for data transmission.

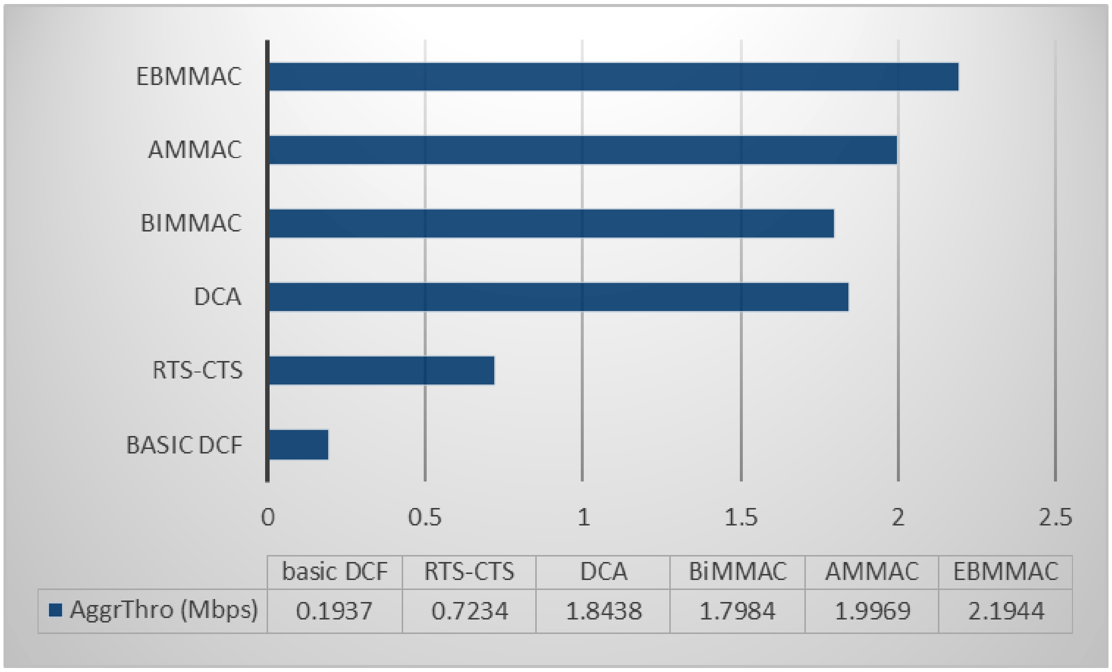


Figure 7.2: Aggregate Throughput Comparison for 500 nodes

Figure 7.2 shows the aggregate throughput for 500 nodes. As we increase the number of nodes, the performance of basic DCF (0.19 Mbps) decreases drastically. RTS-CTS also suffer from degradation in throughput (0.72 Mbps), whereas all the multi-channel protocols are able to sustain the aggregate throughput without great changes. The resulting aggregate throughput for 20, 50, 100, and 500 nodes for all the protocols is shown in Figure 7.3. We can infer that the protocols exhibit a similar trend for all the four different network sizes.

The effective increase (%) of ABMMAC aggregate throughput over other protocols is tabulated in Table 7.1. As we can see, the multi-channel protocols gain a substantial throughput compared to the single channel counterparts by virtue

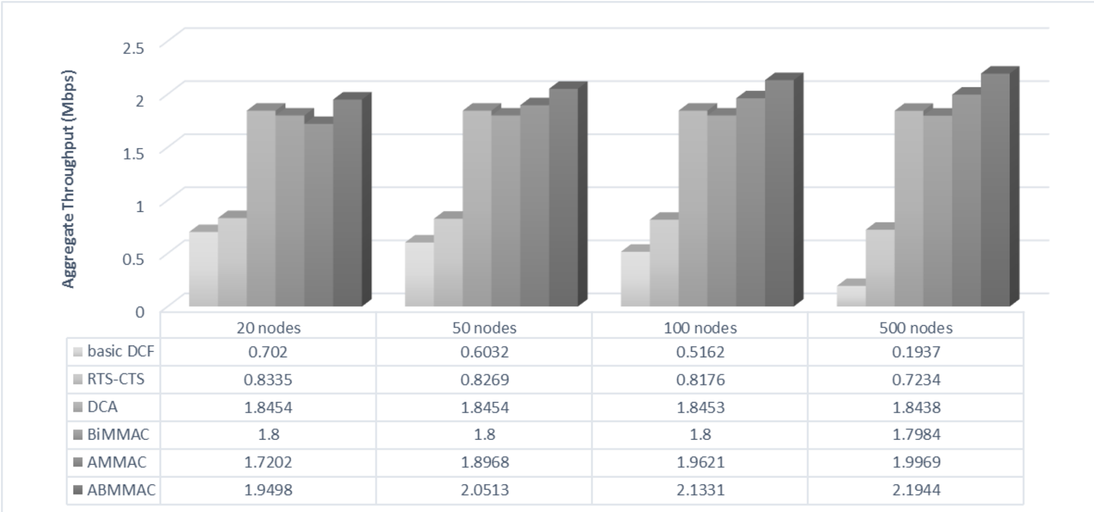


Figure 7.3: Comparison of Aggregate Throughput

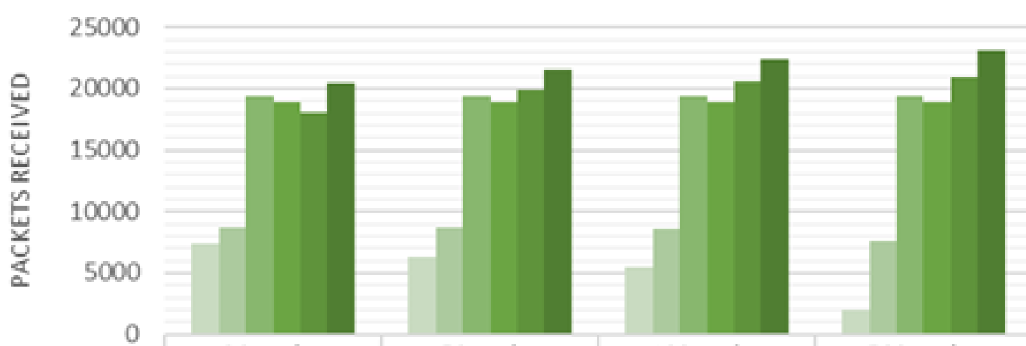
Table 7.1: Effective increase (%) of ABMMAC aggregate throughput over other protocols

Nodes	Basic DCF	RTS-CTS	DCA	BiMMAC	AMMAC
20	177.7493	133.9292	5.6573	8.3222	13.3473
50	240.0696	148.0711	11.1575	13.9611	8.1453
100	313.2313	160.8977	15.5964	18.5056	8.7151
500	1032.886	203.3453	19.0151	22.0196	9.8903

of use of additional data channels. DCA and AMMAC are constrained by the fact of having dedicated a channel for control signal. But given their available data channels, the performance is good. AMMAC performs better than DCA and BiMMAC due to the reuse of control channel. By having an efficient multiplex of AMMAC and BiMMAC, our ABMMAC achieves good throughput over all the other protocols.

7.2 Packets Received

We consider the total number of successful packets received for different network sizes. The results are tabulated in Figure 7.4. As we can infer, multi-channel protocols receive more packets than single channel protocols. In multi-channel protocols, when one channel is occupied, nodes can still transmit data on other available channels. Multiple data frames sent in parallel increase the network throughput.



	20 nodes	50 nodes	100 nodes	500 nodes
basic DCF	7375	6337	5423	2035
RTS-CTS	8757	8687	8590	7600
DCA	19387	19387	19386	19371
BiMMAC	18910	18910	18910	18894
AMMAC	18072	19927	20614	20979
ABMMAC	20484	21551	22410	23054

Figure 7.4: Comparison of Packets Received

Table 7.2 shows the number of packets received more in ABMMAC, in comparison to other protocols. These data are collected for a time period of 10000 data frames. For a network of 20 nodes, ABMMAC received 2412 packets more than AMMAC, and 13019 packets more than basic DCF. For 500 nodes, ABMMAC received 2075 packets more than AMMAC, and 21019 packets more than basic

Table 7.2: Extra Packets Received in ABMMAC in comparison to other protocols

Nodes	Basic DCF	RTS-CTS	DCA	BiMMAC	AMMAC
20	13 109	11 727	1097	1574	2412
50	15 214	12 864	2164	2641	1624
100	16 987	13 820	3024	3500	1796
500	21 019	15 454	3683	4160	2075

DCF. ABMMAC outperforms other multi-channel and single channel protocols, in small and large size networks.

7.3 Throughput Efficiency

The throughput efficiency indicates the effective utilization of the channels. Table 7.3 shows the throughput efficiency of the protocols for network sizes of 20, 50, 100, and 500. A total of three channels are used (two data channels and a control channel).

For small network size (20 nodes), RTS-CTS uses 83.35% of the channel, while basic DCF uses 70.2%. DCA, BiMMAC, AMMAC, and ABMMAC, use 61.51%, 60%, 57.34%, 64.99% respectively. As we can infer, for small network sizes, single channel protocols outperform multi-channel protocols. Multi channel protocols DCA and BiMMAC lose significant bandwidth (33.33%) to control signalling and only 66.67% is left for data transfer. In AMMAC and ABMMAC, nodes that completed data transfer wait for one maximum data frame time before they start another negotiation. This waiting time overhead reduces their throughput efficiency. Nonetheless, ABMMAC has better throughput efficiency compared to

Table 7.3: Throughput Efficiency

Nodes	Basic DCF	RTS-CTS	DCA	BiMMAC	AMMAC	ABMMAC
20	0.7020	0.8335	0.6151	0.6	0.5734	0.6499
50	0.6032	0.8269	0.6151	0.6	0.6323	0.6838
100	0.5162	0.8176	0.6151	0.6	0.6540	0.7110
500	0.1937	0.7234	0.6146	0.5995	0.6656	0.7315

other multi-channel protocols.

When network size increases to 500 nodes, the throughput efficiency of basic DCF degrades to 19.37% due to strong contention among the nodes and increased data collisions. In RTS-CTS, the collisions are limited to shorter control frames. Nonetheless, the efficiency decreases to 72.34%. On the other hand, the multi-channel protocols sustain their throughput efficiency for small and big size networks. ABMMAC outperforms all other protocols even as network size increases to 500 nodes.

7.4 Queueing Delay

As network size increases, the contention among the nodes increases leading to more collisions. Nodes then increase their contention window and attempt to retransmit. On reaching the maximum retransmission limit, the frame is dropped. In general, queueing delay increases directly with the increase in network size.

The queueing delay for 100 nodes is shown in Figure 7.5. Single channel protocols basic DCF, and RTS-CTS suffer a delay of 1081.347 ms and 655.496 ms respectively. Multichannel protocols DCA, BiMMAC, and AMMAC have delays of 435.534 ms, 446.399 ms, and 409.066 ms respectively. ABMMAC has a delay of 375 ms, which is on average 25 ms less than other multi-channel protocols and about 300 to 600 ms less than single channel protocols.

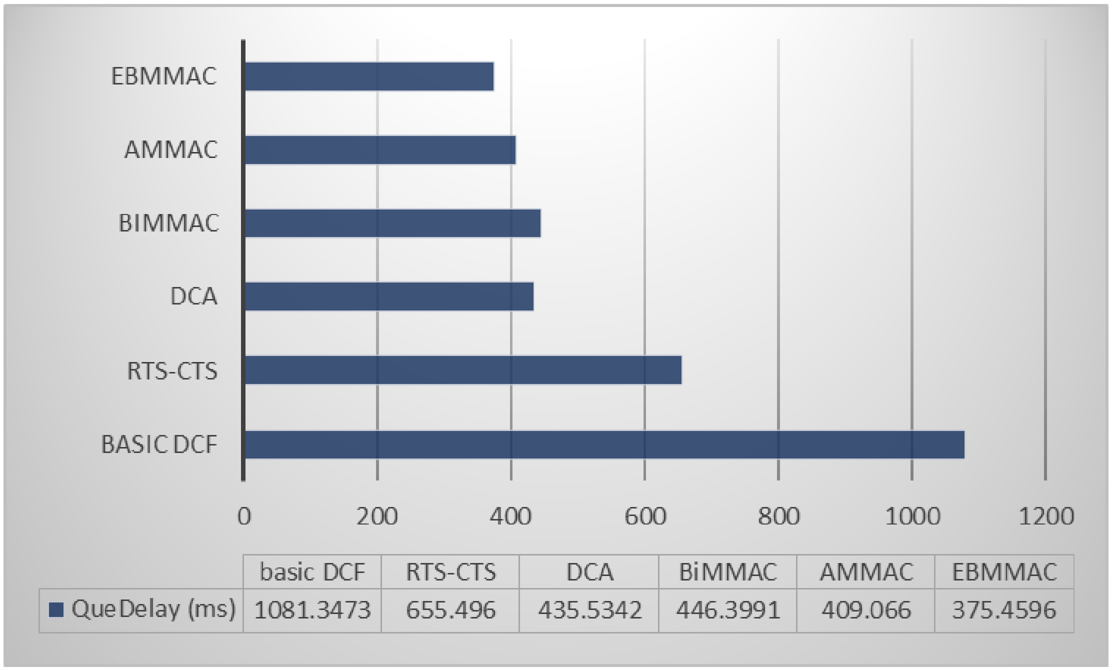


Figure 7.5: Comparison of Queueing Delay for 100 nodes

Figure 7.6 shows the queueing delay for 500 nodes. For basic DCF, the queueing delay grows drastically to 3258.88 ms, while RTS-CTS has delay of about 912.41 ms. DCA, BiMMAC and AMMAC protocols have size-able delays of 2113.754 ms, 2232.618 ms, 2013.946 ms respectively. ABMMAC also experience similar queueing delay (1834.094). However, when compared to AMMAC and BiMMAC, it achieves a reduction of about 200 ms.

The queueing delays for the protocols for 20, 50, 100, and 500 nodes is presented in Figure 7.7. This too shows that the multi-channel protocols gain a good delay reduction compared to their single channel protocols. AMMAC suffers high

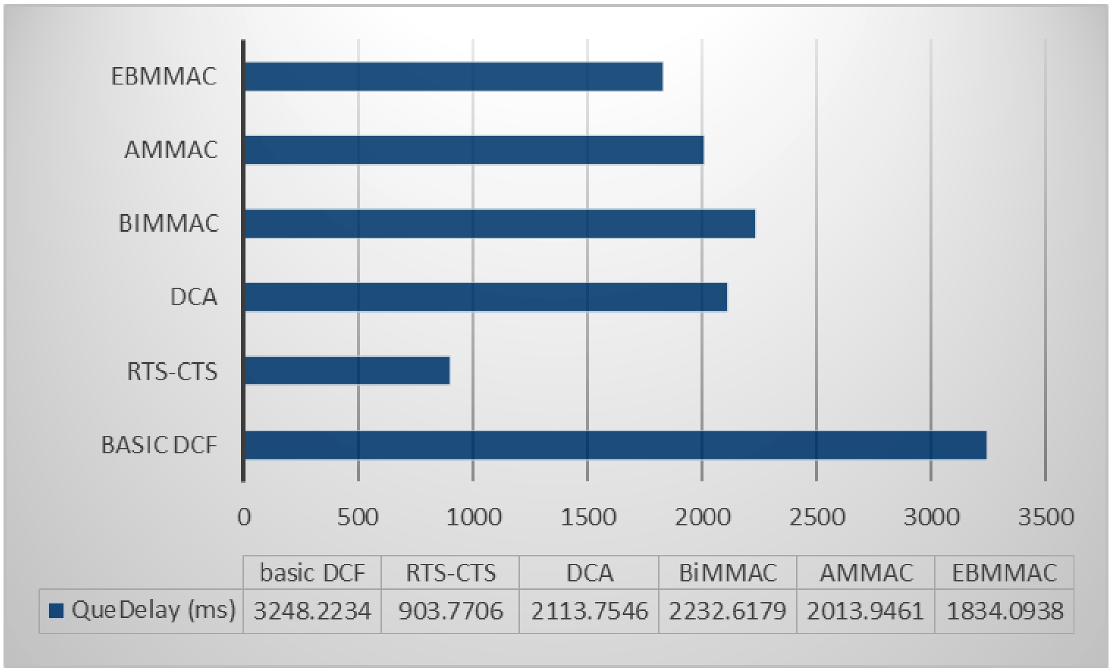


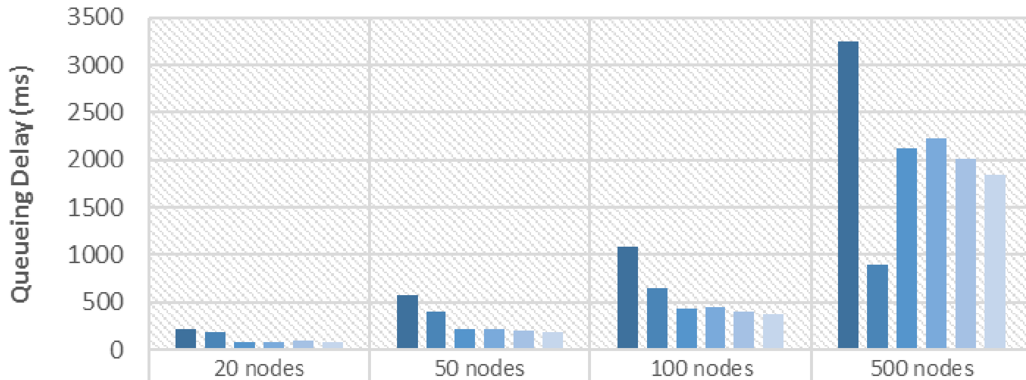
Figure 7.6: Comparison of Queueing Delay (QueDelay) for 500 nodes

delay since control channel is occupied for data transmissions. Other nodes need to wait until the data exchange is over, even though the data channels are available. ABMMAC compensates this delay by allowing two frame transmissions from sender as well as receiver, as that of BiMMAC.

The effective reduction in delay for ABMMAC over the single channel and the multi-channel protocols is tabulated in Table 7.4. From this results, we can observe that ABMMAC exhibits a similar delay as that of BiMMAC, while performing much better than AMMAC.

Table 7.4: Delay Reduction of ABMMAC over other protocols

Nodes	Basic DCF	RTS-CTS	DCA	BiMMAC	AMMAC
20	147.4992	109.6589	4.7853	7.0361	11.2617
50	374.3765	212.5923	22.3663	27.8841	16.3186
100	705.8877	280.0364	60.0746	70.9395	33.6064
500	1414.13	−930.323	279.6608	398.5241	179.8523



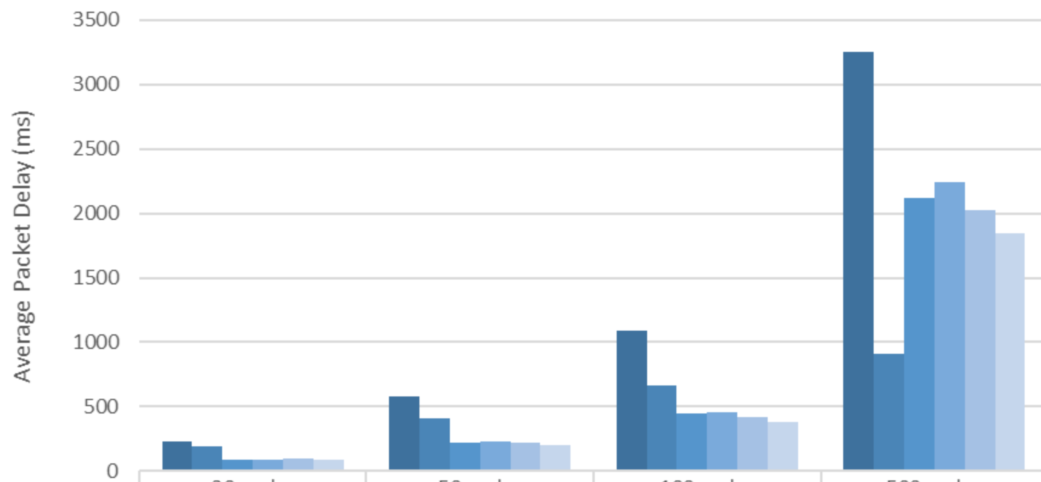
	20 nodes	50 nodes	100 nodes	500 nodes
basic DCF	222.8064	565.5755	1081.3473	3248.2234
RTS-CTS	184.9661	403.7913	655.496	903.7706
DCA	80.0925	213.5653	435.5342	2113.7546
BiMMAC	82.3433	219.0831	446.3991	2232.6179
AMMAC	86.5689	207.5176	409.066	2013.9461
ABMMAC	75.3072	191.199	375.4596	1834.0938

Figure 7.7: Comparison of Queueing Delay

RTS-CTS, by exception, has 930.323ms less delay than ABMMAC. The reason for this is explained in Section 7.6.

7.5 Average Packet Delay

The average packet delay is sum of queueing delay and transmission delay. So, the packet delay follows a very similar trend as that of queueing delay, as shown in Figure 7.8.

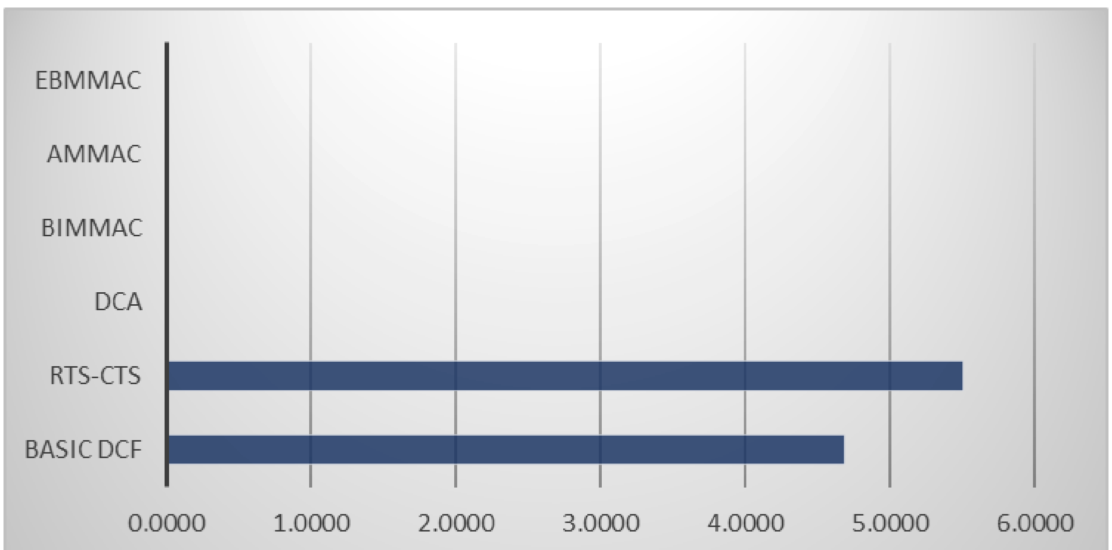


	20 nodes	50 nodes	100 nodes	500 nodes
basic DCF	231.4464	574.2155	1089.9873	3256.8634
RTS-CTS	193.6061	412.4313	664.136	912.4106
DCA	88.7325	222.2053	444.1742	2122.3946
BiMMAC	90.9833	227.7231	455.0391	2241.2579
AMMAC	95.2089	216.1576	417.706	2022.5861
ABMMAC	83.9472	199.839	384.0996	1842.7338

Figure 7.8: Comparison of Average Packet Delay

7.6 Frame Drop Ratio

The frame drop ratio for the protocols for 100 and 500 nodes is shown in Figures 7.9 and 7.10. The single channel protocols experience a frame drop ratio at about 5% for 100 nodes. As the number of nodes grows to 500, the frame drop ratio for both basic DCF and RTS-CTS increases to 60%. Irrespective of usage of RTS-CTS, the single channel protocols suffer high performance degradation as the network size grows, as they use only one channel for data transmission. Multichannel protocols have a nill frame drop for 100 nodes. And for 500 nodes, AMMAC has 0.48% drop ratio, DCA 1.19% and ABMMAC 0.238%. The results suggest that multi-channel protocols have very negligible frame drops even as the size of the network increases.



	basic DCF	RTS-CTS	DCA	BiMMAC	AMMAC	EBMMAC
FDR	4.6924	5.5006	0	0	0	0

Figure 7.9: Comparison of Frame Drop Ratio for 100 nodes

The frame drop ratios for the protocols for network size of 20, 50, 100, and

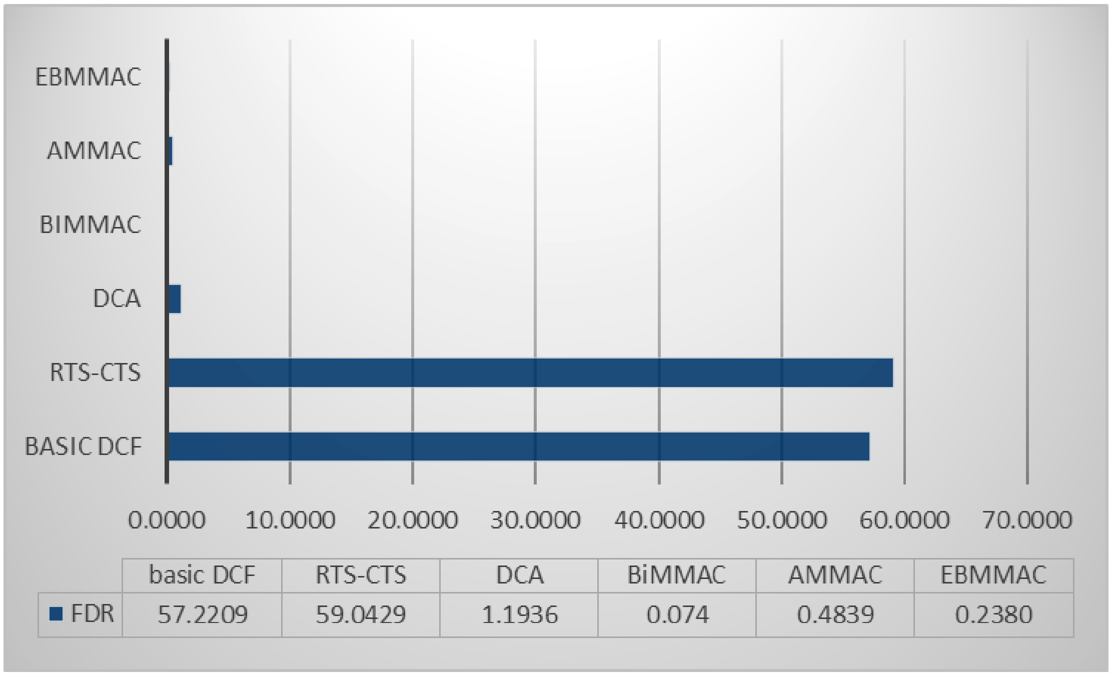


Figure 7.10: Comparison of Frame Drop Ratio for 500 nodes

500 nodes is shown in Figure 7.11.

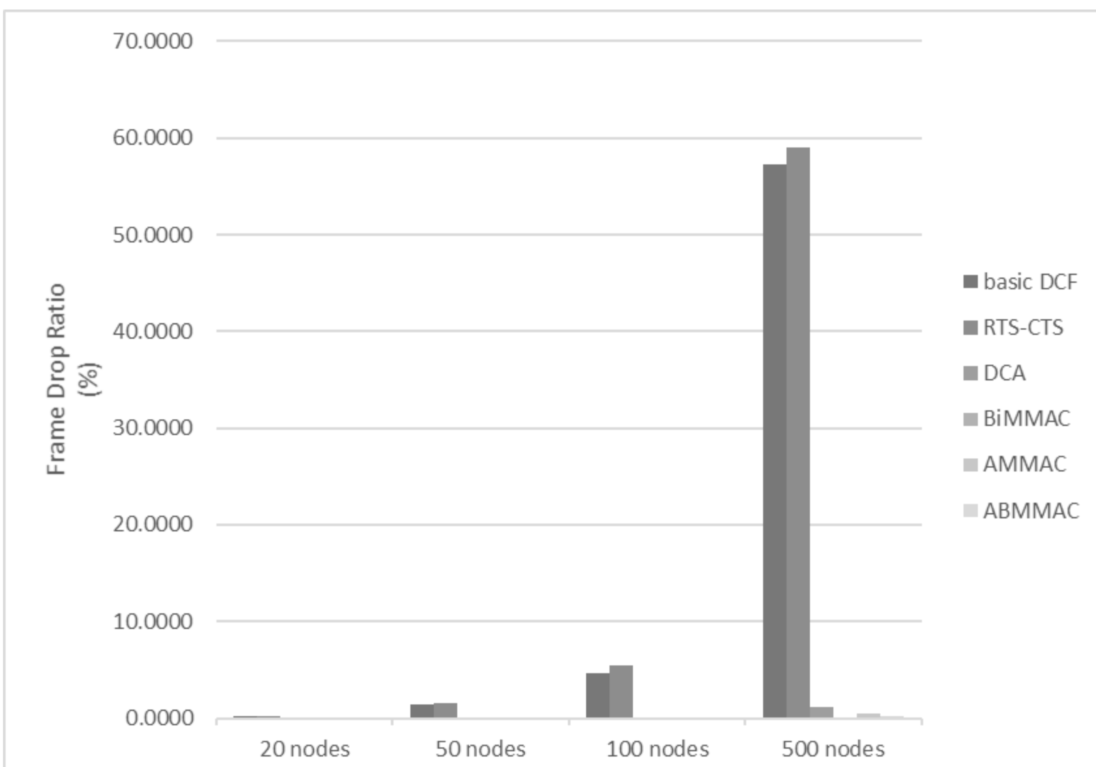


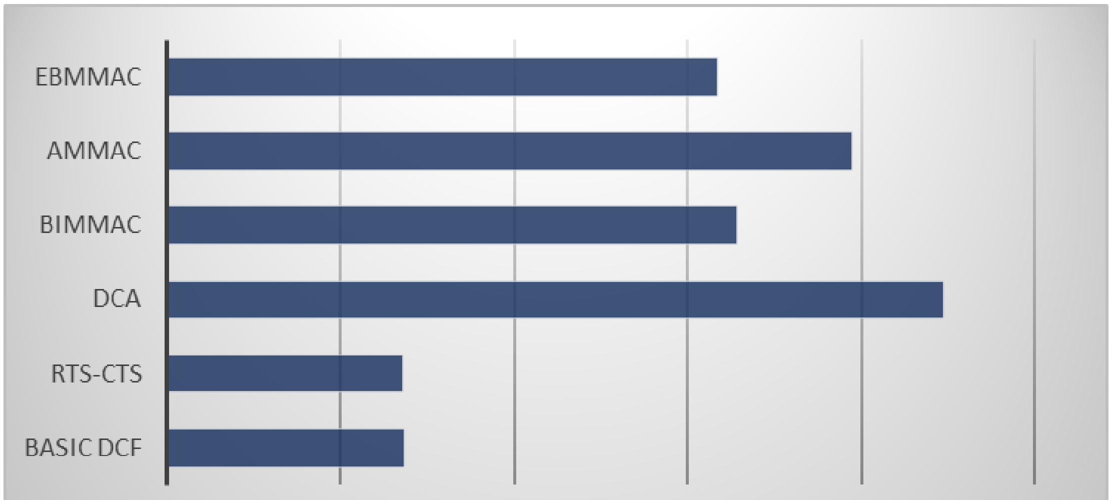
Figure 7.11: Comparison of Frame Drop Ratio

In simulation we observed that, for the scenario of 500 nodes, basic DCF sent 2722 packets successfully out of 4757, whereas RTS-CTS sent 10956 packets successfully out of 18556. As seen in Figure 7.4 basic DCF dropped 2722 packets while RTS-CTS dropped 10956 packets. The frame drop ratio is almost similar for basic DCF (57%)and RTS-CTS (59%). This suggests that RTS-CTS manages to send lot more packets than basic DCF and also drops a size-able ratio. Since dropped packets are not considered in our calculation of queueing delays, RTS-CTS shows 930.323 reduced delay than ABMMAC.

7.7 Jain Fairness Index

In saturated traffic condition, every node has a packet to transmit all the time. JFI can indicate how much of the bandwidth is allotted per node.A window size of 1 is considered and this is normalized to the number of nodes present. For example, for 100 nodes, the channel trace of 100 successful accesses is taken and JFI is calculated based on a sliding window method is used similar to that of [1]. JFI of 1 indicates a maximum fairness and value of $\frac{1}{N}$ indicates the minimum, where N is the total number of nodes.

The Jain Fairness Index comparison for 100 nodes is presented in Figure 7.12. JFI for single channel protocols is about 0.25. The underlying Binary Exponential Backoff (BEB) algorithm in 802.11 gives way to unfairness issue [50]. It favours the successful nodes by allowing a minimum back off and collided nodes to have a longer waiting time.



	basic DCF	RTS-CTS	DCA	BiMMAC	AMMAC	EBMMAC
JFI	0.2742	0.2728	0.8954	0.6566	0.7898	0.6348

Figure 7.12: Comparison of Jain Fairness Index for 100 nodes

As multi-channel protocols have greater scope for transmissions, the nodes can get access to data channels easily and as such, the collisions are greatly reduced. DCA has JFI of about 0.895 and AMMAC about 0.79. BiMMAC is not built for fairness but to increase the network throughput by allowing receivers to take privilege of successful handshake and send a data frame. As such other nodes need to wait for longer time and JFI drops to 0.657. ABMMAC also has a JFI of 0.65 which is expected since its operation on data channels follows BiMMAC.

In JFI for 500 nodes, as shown in Figure 7.13, basic DCF and RTS-CTS improved a bit of its fairness index to 0.4. The dropping of packets in relation to increased network size seems to have an impact on fairness index for single channel protocols. For multi channel protocols, AMMAC, BiMMAC, and ABMMAC maintain about the same fairness index at 0.79, 0.64, and 0.63 respectively. How-

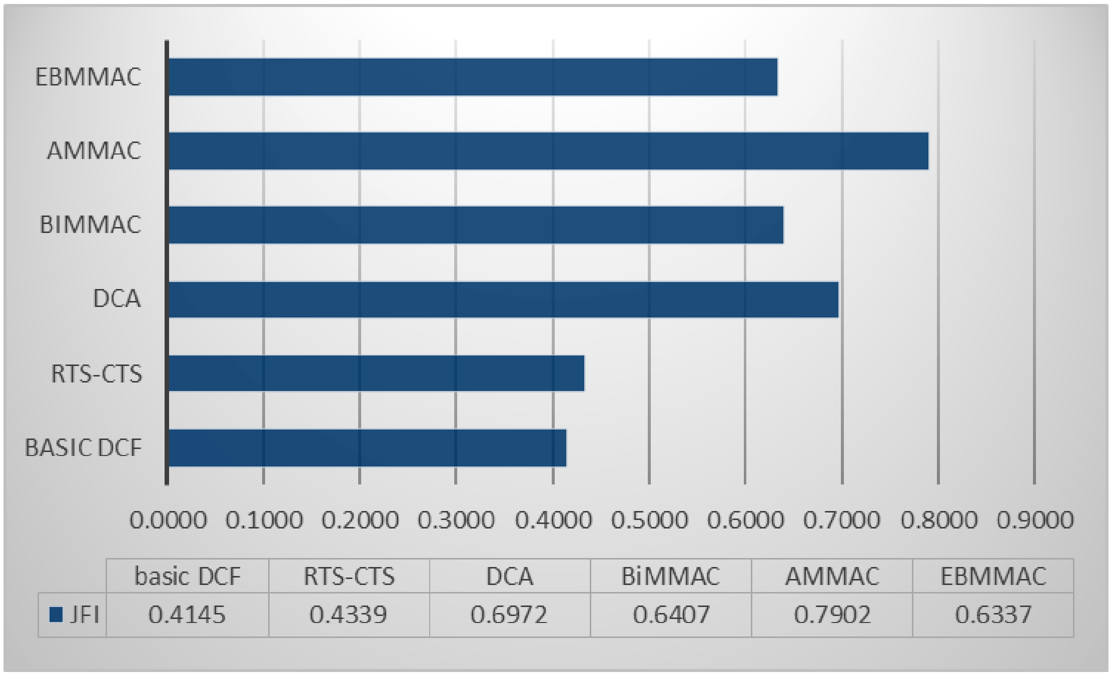


Figure 7.13: Comparison of Jain Fairness Index for 500 nodes

ever, JFI of DCA drops to 0.697. This is due to the saturation on the control channel. As the number of nodes grow to 500, more collisions and backoff happen on the control channel and it becomes an overhead. So, DCA cannot sustain its high fairness index for large networks.

CHAPTER 8

LIMITATIONS AND FUTURE WORK

The dedicated control channel approaches have a problem in accommodating higher number of channels due to problem of control channel saturation. Since ABMMAC use a common channel for control signalling and reuse that channel for data transfer, this protocol can suffer when the number of data channels increase, for example, 802.11a. The evaluation of the selected protocols is done for saturated traffic conditions for a single hop network. The mobility of the nodes and multi-hop scenario are not considered and we leave this for future study.

CHAPTER 9

CONCLUSION

We proposed an asynchronous bidirectional multi-channel MAC (ABMMAC) for 802.11 networks with evaluation of factors such as throughput efficiency, queueing delay, frame drop ratio, and fairness index. The protocol is compatible to legacy 802.11, uses asynchronous mode of operation, by just using a single half-duplex radio. By reuse of control channel and allowing two frames transmissions on data channels, the proposed protocol outperforms its multi-channel variants and single channel 802.11. For big network sizes, the protocol gives 200-1000% better throughput, 1000ms less delay than single channel protocol and 10-20% better throughput, 200ms less delay than its multi-channel variants. The protocol is tested by reliable simulations and can be easily deployed for 802.11b networks.

REFERENCES

[1] A. Al-Akeel, "Optimizing backoff procedure for enhanced throughput and fairness in wireless lans," Master's thesis, King Fahd University of Petroleum & Minerals, 2007.

[2] I. F. Akyildiz and X. Wang, "Cross-layer design in wireless mesh networks," *Vehicular Technology, IEEE Transactions on*, vol. 57, no. 2, pp. 1061–1076, 2008.

[3] "Ieee standard 802.11: wireless lan medium access control (mac) and physical layer (phy) specifications," *New York: IEEE*, 1999.

[4] B. Han, "Performance evaluation of channel selection algorithm for multi-channel mac protocol in ad hoc networks," 2012.

[5] S.-C. Lo, "Design of multichannel mac protocols for wireless ad hoc networks," *International Journal of Network Management*, vol. 19, no. 5, pp. 399–413, 2009.

[6] S.-L. Wu, C.-Y. Lin, Y.-C. Tseng, and J.-P. Sheu, "A new multi-channel mac protocol with on-demand channel assignment for multi-hop mobile ad

hoc networks," in *Parallel Architectures, Algorithms and Networks, 2000. I-SPAN 2000. Proceedings. International Symposium on.* IEEE, 2000, pp. 232–237.

[7] J. So and N. H. Vaidya, "Multi-channel mac for ad hoc networks: handling multi-channel hidden terminals using a single transceiver," in *Proceedings of the 5th ACM international symposium on Mobile ad hoc networking and computing.* ACM, 2004, pp. 222–233.

[8] I. Wormsbecker and C. Williamson, "On channel selection strategies for multi-channel mac protocols in wireless ad hoc networks," in *Wireless and Mobile Computing, Networking and Communications, 2006.(WiMob'2006). IEEE International Conference on.* IEEE, 2006, pp. 212–220.

[9] T. Kuang and C. Williamson, "A bidirectional multi-channel mac protocol for improving tcp performance on multihop wireless ad hoc networks," in *Proceedings of the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems.* ACM, 2004, pp. 301–310.

[10] A. Arafeh, F. Oughali, T. Sheltami, A. Mahmoud, and K. Daabaj, "A contention free multi-channel mac protocol with improved negotiation efficiency for wireless ad-hoc networks," 2010.

[11] J. So and N. H. Vaidya, "A routing protocol for utilizing multiple channels in multi-hop wireless networks with a single transceiver," *University of Illinois at Urbana-Champaign*, 2004.

[12] R. Vedantham, S. Kakumanu, S. Lakshmanan, and R. Sivakumar, "Component based channel assignment in single radio, multi-channel ad hoc networks," in *Proceedings of the 12th annual international conference on Mobile computing and networking*. ACM, 2006, pp. 378–389.

[13] R. Soua and P. Minet, "Multichannel assignment protocols in wireless sensor networks: A comprehensive survey," *Pervasive and Mobile Computing*, vol. 16, pp. 2–21, 2015.

[14] M. Doudou, D. Djenouri, N. Badache, and A. Bouabdallah, "Synchronous contention-based mac protocols for delay-sensitive wireless sensor networks: A review and taxonomy," *Journal of Network and Computer Applications*, vol. 38, pp. 172–184, 2014.

[15] K. H. Almotairi and X. S. Shen, "A distributed multi-channel mac protocol for ad hoc wireless networks," *Mobile Computing, IEEE Transactions on*, vol. 14, no. 1, pp. 1–13, 2015.

[16] H. Lei, Z. Ren, C. Gao, and Y. Guo, "A new multi-channel mac protocol for 802.11-based wireless mesh networks," in *Computer Science and Electronics Engineering (ICCSEE), 2012 International Conference on*, vol. 1. IEEE, 2012, pp. 27–31.

[17] A. Ali, H. Wang, H. Lv, and X. Chen, "Asynchronous session reservation multi-channel mac for ad hoc networks," *Journal of Networks*, vol. 9, no. 2, pp. 259–268, 2014.

[18] C. Chereddi, P. Kyasanur, and N. H. Vaidya, "Net-x: a multichannel multi-interface wireless mesh implementation," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 11, no. 3, pp. 84–95, 2007.

[19] C. Cetinkaya, "Asynchronous multi-channel mac protocol," *Journal of Networks*, vol. 10, no. 9, pp. 530–542, 2016.

[20] J. Shi, T. Salonidis, and E. W. Knightly, "Starvation mitigation through multi-channel coordination in csma multi-hop wireless networks," in *Proceedings of the 7th ACM international symposium on Mobile ad hoc networking and computing.* ACM, 2006, pp. 214–225.

[21] M. Ramakrishnan and P. V. Ranjan, "Multi channel mac implementation for wireless sensor networks," in *Advances in Computing, Control, & Telecommunication Technologies, 2009. ACT'09. International Conference on.* IEEE, 2009, pp. 809–813.

[22] T. Luo, M. Motani, and V. Srinivasan, "Cooperative asynchronous multichannel mac: Design, analysis, and implementation," *Mobile Computing, IEEE Transactions on*, vol. 8, no. 3, pp. 338–352, 2009.

[23] M. Sarkar, S. Nagaraj, and I. H. Balsania, "A sinr based mac layer protocol for multi-channel ad-hoc networks," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2011 7th International.* IEEE, 2011, pp. 1889–1893.

[24] K. H. Almotairi and X. Shen, "Mmac-hr: Multi-channel medium access control with hopping reservation for multi-hop wireless networks," in *Global Telecommunications Conference (GLOBECOM 2010), 2010 IEEE*. IEEE, 2010, pp. 1–5.

[25] B. Yang, B. Li, Z. Yan, M. Yang, and X. Zuo, "A reliable channel reservation based multi-channel mac protocol with a single transceiver," in *Heterogeneous Networking for Quality, Reliability, Security and Robustness (QSHINE), 2015 11th International Conference on*. IEEE, 2015, pp. 265–271.

[26] W.-H. Liao and W.-C. Chung, "An efficient multi-channel mac protocol for mobile ad hoc networks," in *Communications and Mobile Computing, 2009. CMC'09. WRI International Conference on*, vol. 2. IEEE, 2009, pp. 162–166.

[27] D. N. M. Dang, M. Van Nguyen, C. S. Hong, S. Lee, and K. Chung, "An energy efficient multi-channel mac protocol for wireless ad hoc networks," in *Global Communications Conference (GLOBECOM), 2012 IEEE*. IEEE, 2012, pp. 433–438.

[28] W.-T. Chen, J.-C. Liu, T.-K. Huang, and Y.-C. Chang, "Tammac: an adaptive multi-channel mac protocol for manets," *Wireless Communications, IEEE Transactions on*, vol. 7, no. 11, pp. 4541–4545, 2008.

[29] W.-H. Liao, K.-P. Shih, and W.-C. Chung, "Multi-channel medium access control protocol with channel distribution for mobile ad hoc networks," *Communications, IET*, vol. 3, no. 12, pp. 1821–1831, 2009.

[30] D. N. M. Dang and C. S. Hong, "H-mmac: A hybrid multi-channel mac protocol for wireless ad hoc networks," in *Communications (ICC), 2012 IEEE International Conference on.* IEEE, 2012, pp. 6489–6493.

[31] M. Won, C. Yang, W. Zhou, and R. Stoleru, "Energy efficient multi-channel media access control for dense wireless ad hoc and sensor networks," *Wireless networks*, vol. 19, no. 7, pp. 1537–1551, 2013.

[32] A. Tzamaloukas and J. Garcia-Luna-Aceves, "Channel-hopping multiple access," in *Communications, 2000. ICC 2000. 2000 IEEE International Conference on*, vol. 1. IEEE, 2000, pp. 415–419.

[33] C.-T. Chang, C.-Y. Chang, and Y.-J. Lu, "Ecu-mac: A multi-channel mac protocol for enhancing channel utilization in wireless networks," in *Ubiquitous and Future Networks (ICUFN), 2012 Fourth International Conference on.* IEEE, 2012, pp. 52–57.

[34] K.-P. Shih, Y.-D. Chen, and S.-S. Liu, "A collision avoidance multi-channel mac protocol with physical carrier sensing for mobile ad hoc networks," in *2010 IEEE 24th International Conference on Advanced Information Networking and Applications Workshops.* IEEE, 2010, pp. 656–661.

[35] C.-M. Chao, H.-C. Tsai, and C.-Y. Huang, "Load-aware channel hopping protocol design for mobile ad hoc networks," in *Wireless and Pervasive Computing (ISWPC), 2012 7th International Symposium on.* IEEE, 2012, pp. 1–6.

[36] P. Bahl, R. Chandra, and J. Dunagan, "Ssch: slotted seeded channel hopping for capacity improvement in ieee 802.11 ad-hoc wireless networks," in *Proceedings of the 10th annual international conference on Mobile computing and networking*. ACM, 2004, pp. 216–230.

[37] W. So, J. Walrand, J. Mo *et al.*, "Mcmac: A parallel rendezvous multi-channel mac protocol," in *Wireless Communications and Networking Conference, 2007. WCNC 2007. IEEE*. IEEE, 2007, pp. 334–339.

[38] L. Tang, Y. Sun, O. Gurewitz, and D. B. Johnson, "Em-mac: a dynamic multichannel energy-efficient mac protocol for wireless sensor networks," in *Proceedings of the Twelfth ACM International Symposium on Mobile Ad Hoc Networking and Computing*. ACM, 2011, p. 23.

[39] C.-M. Chao, H.-C. Tsai, and K.-J. Huang, "A new channel hopping mac protocol for mobile ad hoc networks," in *Wireless Communications & Signal Processing, 2009. WCSP 2009. International Conference on*. IEEE, 2009, pp. 1–5.

[40] J. Mo, H.-S. W. So, and J. Walrand, "Comparison of multichannel mac protocols," *Mobile Computing, IEEE Transactions on*, vol. 7, no. 1, pp. 50–65, 2008.

[41] D. Nguyen, J. Garcia-Luna-Aceves, and K. Obraczka, "Collision-free asynchronous multi-channel access in ad hoc networks," in *Global Telecommu-*

nications Conference, 2009. GLOBECOM 2009. IEEE. IEEE, 2009, pp. 1–6.

[42] H. Karl and A. Willig, *Protocols and architectures for wireless sensor networks.* John Wiley & Sons, 2007.

[43] P. Chatzimisios, V. Vitsas, and A. C. Boucouvalas, "Throughput and delay analysis of ieee 802.11 protocol," in *Networked Appliances, 2002. Liverpool. Proceedings. 2002 IEEE 5th International Workshop on.* IEEE, 2002, pp. 168–174.

[44] S.-T. Sheu, T. Chen, J. Chen, and F. Ye, "The impact of rts threshold on ieee 802.11 mac protocol," in *Parallel and Distributed Systems, 2002. Proceedings. Ninth International Conference on.* IEEE, 2002, pp. 267–272.

[45] G. Bianchi, L. Fratta, and M. Oliveri, "Performance evaluation and enhancement of the csma/ca mac protocol for 802.11 wireless lans," in *Personal, Indoor and Mobile Radio Communications, 1996. PIMRC'96., Seventh IEEE International Symposium on*, vol. 2. IEEE, 1996, pp. 392–396.

[46] G. Bianchi, "Ieee 802.11-saturation throughput analysis," *Communications Letters, IEEE*, vol. 2, no. 12, pp. 318–320, 1998.

[47] H. Wu, S. Cheng, Y. Peng, K. Long, and J. Ma, "Ieee 802.11 distributed coordination function (dcf): analysis and enhancement," in *Communications, 2002. ICC 2002. IEEE International Conference on*, vol. 1. IEEE, 2002, pp. 605–609.

[48] R. Jain, A. Durresi, and G. Babic, "Throughput fairness index: An explanation," Tech. rep., Department of CIS, The Ohio State University, Tech. Rep., 1999.

[49] G. Bianchi, "Performance analysis of the ieee 802.11 distributed coordination function," *Selected Areas in Communications, IEEE Journal on*, vol. 18, no. 3, pp. 535–547, 2000.

[50] S. Xu and T. Saadawi, "Does the ieee 802.11 mac protocol work well in multihop wireless ad hoc networks?" *Communications Magazine, IEEE*, vol. 39, no. 6, pp. 130–137, 2001.

Vitae

- Name: Abdullah Devendiran
- Nationality: Indian
- Date of Birth: 27.01.1985
- Email: *hpzabdullah@gmail.com*
- Permanent Address: Tamilnadu, India
- Publications: A paper out of this thesis is waiting to be submitted.
- Short Biography: I completed Bachelor's Degree in Information Technology at Madras Institute of Technology, Anna University, Chennai, India in the year 2006. I am privileged to have worked in Hewlett Packard and Verizon in Java/J2EE/Unix application development and support.